SBC 211/212
DISKETTE HARDWARE SYSTEM
HARDWARE REFERENCE MANUAL

Manual Order Number: 98-349B

SBC 211/212
DISKETTE HARDWARE SYSTEM
HARDWARE REFERENCE MANUAL

Manual Order Number: 98-349B

Intel Corporation, 3065 Bowers Avenue, Santa Clara, California 95051

PREFACE

This reference manual is intended as the primary source of information on the Intel Diskette Hardware System. We have tried to explain, in an easy to follow format, how to incorporate the diskette system into an Intel OEM computer system, how it communicates with the other SBC products and what internal operations take place.

Refer to the SBC 915 User's Guide for a description of the optional Go/No Go Diagnostic and Monitor Program for SBC 80/10 and System 80/10, and to the SBC 925 User's Guide for a description of the optional Go/No Go Diagnostic and Monitor Program for SBC 80/20 and System 80/20.

TABLE OF CONTENTS

TABLE OF CONTENTS - (Continued)

APPENDICES

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS - (Continued)

LIST OF TABLES

CHAPTER 1
INTRODUCTION

The SBC 211/212 Diskette Hardware System provides a bulk storage capability for Intel's OEM computer systems. The SBC 211 and 212 systems include an intelligent controller (the SBC 201) and up to two diskette drives (refer to Section 1.1). Each drive provides 2,050,048 user-accessible data bits of storage with a data transfer rate of 250,000 bits/second. The controller has been implemented with Intel's powerful Series 3000 Bipolar Computing Elements. The controller provides an interface to the Intel System bus (Multibus) as well as supporting the two diskette drives. The controller records all data in the IBM soft-sectored format, described in Section 1.2.

1.1 SYSTEM OVERVIEW

In addition to one or two diskette drives, their enclosure and power supplies, the Diskette System includes the Channel Board and the Interface Board. These two printed circuit boards constitute what we refer to as the SBC 201 Diskette Controller. The controller plugs into our standard cardcage or your custom backplane. Each of the system components is shown in Figure 1-1, and described in the following paragraphs:

The Channel Board is the primary control module within the Diskette System. The Channel Board receives, decodes and responds to channel commands from a Central Processor Unit (CPU) in the computer system. The Channel Board can access common system memory to determine the particular diskette operations to be performed and to fetch the parameters required for the successful completion of the specified operations. The Channel Board also monitors Diskette System status and error conditions, and organizes these indications into "result type" and "result byte" words that can be read by the OEM computer.

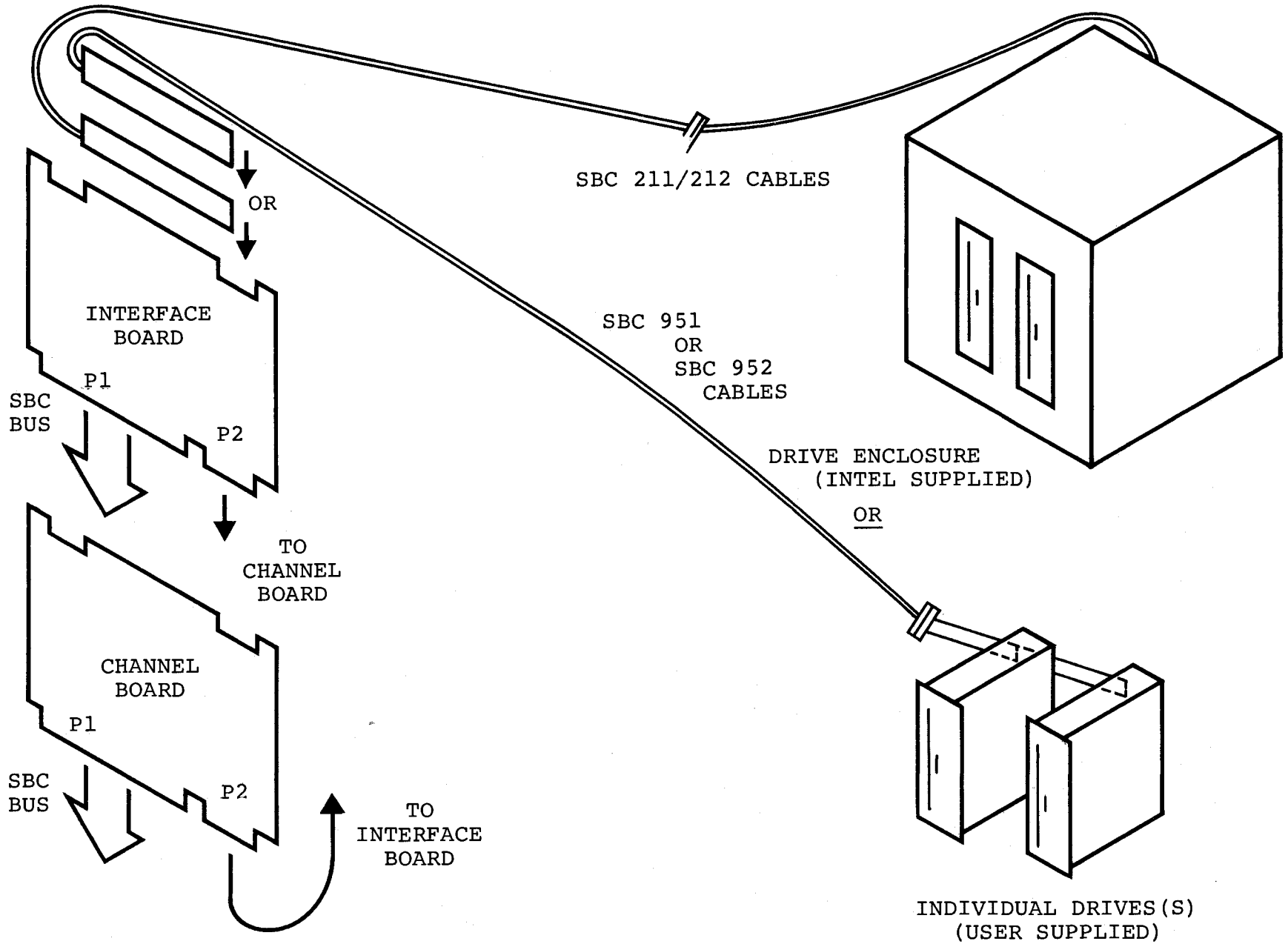


FIGURE 1-1

SBC DISKETTE SYSTEM BLOCK DIAGRAM

The control functions of the Channel and Interface Boards are provided by an 8-bit microprogrammed processor, implemented with Intel's Series 3000 Bipolar Computing Elements. The 8-bit controller includes four 3002 Central Processing Elements (2-bit slice per CPE), a 3001 Microprogram Control Unit and 512 x 32 bits of 3604 programmable-read-only-memory (PROM) which stores the microprogram. The processing and control capabilities of the Diskette system are achieved by execution of the microprogram.

The Interface Board provides the diskette controller with a means of communicating with the diskette drives, as well as with the system bus. Under control of the microprogram executed from the Channel Board, the Interface Board generates those signals which cause the read/write head on the selected drive to be loaded (i.e., to come in contact with the diskette platter), then cause the head to move to the proper track. The Interface Board accepts the data being read off the diskette, interprets certain synchronizing bit patterns, checks the validity of the data using a cyclic redundancy check (CRC) polynomial, and passes the data to the Channel Board.

During write operations, the Interface Board outputs the data and clock bits to the selected drive at the proper times. It also generates CRC characters which are appended to the data; this allows the data to be verified when it is subsequently read.

When the diskette controller requires access to the common system memory, the Interface Board requests and maintains master control of the system bus, and generates the appropriate memory command.

When a CPU in the Intel®OEM computer system issues a channel command to the Diskette System, the Interface Board acknowledges the command as required by system bus protocol.

Each diskette drive consists of read/write and control electronics (on a single printed circuit board), drive mechanism, read/write head, track positioning mechanism and the removable diskette platter. These components interact to perform the following functions:

- Interpret and generate control signals.
- Move read/write head to selected track.
- Read and write data

Table 1-1 lists the performance characteristics for each diskette drive.

TABLE 1-1
DISKETTE DRIVE PERFORMANCE SPECIFICATIONS

Capacity (formatted)	Per Disk -	256,256 bytes
	Per Track -	3,328 bytes
Data Transfer Rate:		250 kilobits/second
Access Time:	Track to Track -	10 ms
	Settling Time -	10 ms
Average Access Time:		260 ms
Rotational Speed:		360 RPM
Average Latency:		83 ms
Recording Mode:		Frequency Modulation

The remaining chapters of the manual deal with each of the system elements in detail. Chapter 2 describes the range of operations that can be performed by the Diskette System, and also provides specific information on how to program the system to execute each of the possible operations. Chapters 3 and 4 provide detailed information on the theory of operation for the Channel Board and the Interface Board, respectively. The final section in each of

these chapters provides a complete schematic drawing of the board as well as a detailed pin list. The reader should continually refer to these schematic drawings in the course of reading the theory of operation sections.

NOTE: To avoid any confusion when referring to the schematics for the Channel and Interface Boards, or when reading the corresponding circuit descriptions, the following notation, concerning the active level of a signal, will apply:

> *Whenever a signal is active-low, its mnemonic is followed by a slash; for example, MRDC/ means that the level on that line will be low when the memory read command is true (active). If the signal is subsequently inverted, thus making it active-high, the slash is omitted; for example, MRDC means that the level on that line will be high when the memory command is true.*

Chapter 5 lists the manufacturer's information on the diskette drives. Chapter 6 provides the major state flow charts for the micro-program which is executed by the Series 3000 Bipolar Microcomputer Set (on the Channel Board), and which essentially controls operation of the Diskette System. Chapter 7 provides basic information on the installation and use of the Diskette System. Finally, Chapter 8 summarizes the AC and DC operating characteristics for the Diskette System.

Before proceeding to Chapter 2, however, we first provide a comprehensive review of the IBM soft-sectored recording format which is used by the Diskette System.

1.2 RECORDING FORMAT

This section summarizes the specifications for the IBM soft-sectored recording format used by the Diskette System.

Physical Data Format:

The physical data format is the format with which the diskette controller circuitry must interact. The elements of the physical data format are the hard index hole, index mark, sector address marks, sector headers, and data sectors. The index mark and sector address marks are recorded with unique clock patterns requiring the controller circuitry to accumulate the unique clock patterns for index and sector address mark identification. Figure 1-2 illustrates the general physical data format.

A "byte", when referring to serial data (being written to or read from the diskette drive), is defined as eight (8) consecutive bit cells. By definition, a bit cell is the period between the leading edge of one clock bit and the leading edge of the next clock bit. The most significant bit cell is defined as bit cell 0 and the least significant bit cell is defined as bit cell 7. When reference is made to a specific data bit (i.e., data bit 3), it is with respect to the corresponding bit cell (bit cell 3).

During a write operation bit cell 0 of each byte is transferred to the drive first with bit cell 7 being transferred last. Correspondingly, the most significant byte of data is transferred to the diskette first and the least significant byte is transferred last.

When data is being read back from the drive, bit cell 0 of each byte will be transferred first with bit cell 7 last. As with reading, the most significant byte will be transferred first from the drive to the user.

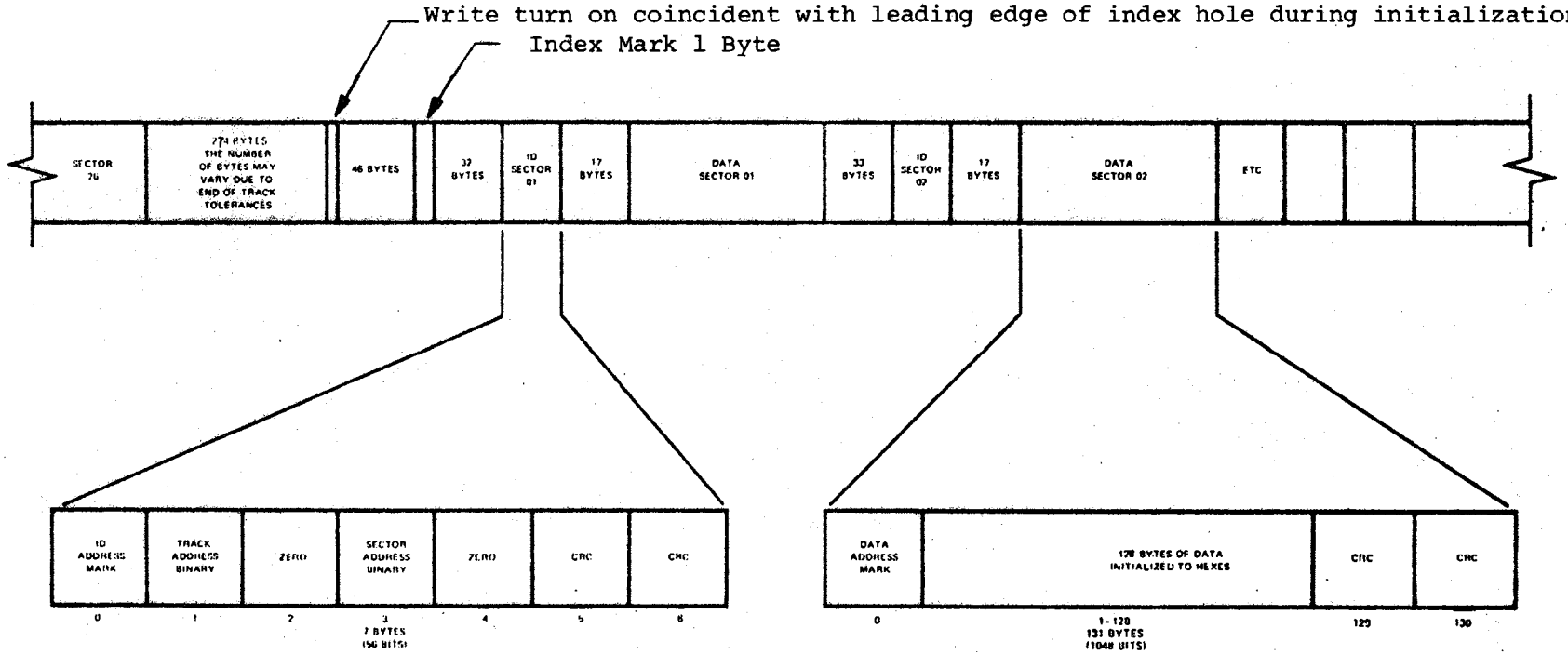


FIGURE 1-2

Physical Data Format

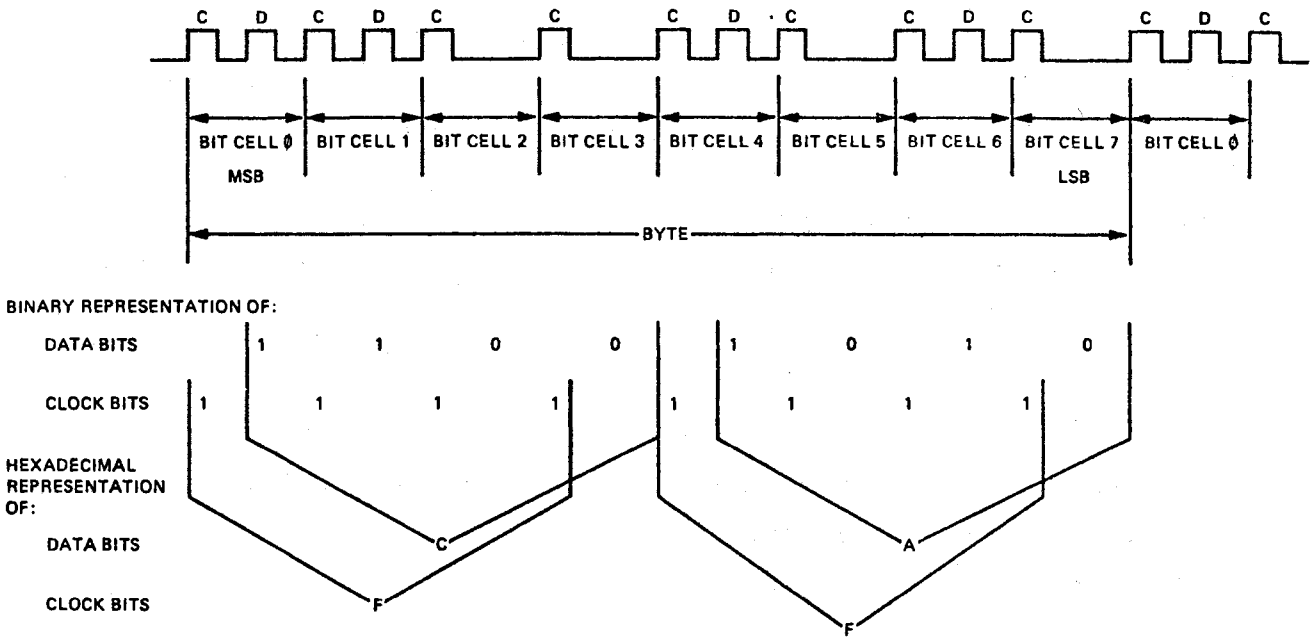


FIGURE 1-3

BYTE REPRESENTATION

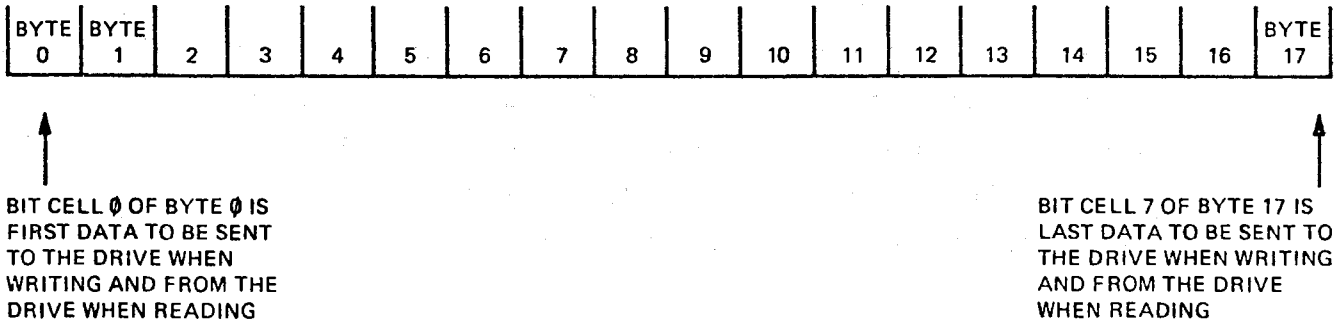


FIGURE 1-4

DATA BYTES

Figure 1-3 illustrates the relationship of the bits within a byte and Figure 1-4 illustrates the relationship of the bytes for read and write data.

Data is recorded on the diskette using frequency modulation as the recording mode; that is, each data bit recorded on the diskette has an associated clock bit recorded with it. Data written on and read back from the diskette takes the form shown in Figure 1-5. As shown in Figure 1-6, the clock bits and data bits (if present) are interleaved.

Track Format:

Each track recorded on a diskette consists of 26 fixed length records along with the necessary gaps for record updating. Figure 1-7 illustrates the format of one complete track.

Each field on a track is separated from adjacent fields by a number of bytes containing no data. These areas are referred to as gaps and are provided to allow the updating of one field without affecting adjacent fields. As can be seen from Figure 1-7, there are four different types of gaps on each track:

- Gap 1 - Post-Index Gap
  This gap is defined as the 32 bytes between Index Address Mark and the ID Address Mark for Sector one (excluding the address mark bytes). This gap is always 32 bytes in length and is not affected by any updating process.

- Gap 2 - ID Gap
  The seventeen bytes between the ID Field and the Data Field are defined as Gap 2 (ID Gap). This gap may vary in size slightly after the Data field has been updated.

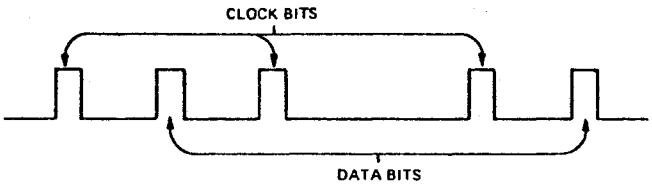


FIGURE 1-5
DATA BIT

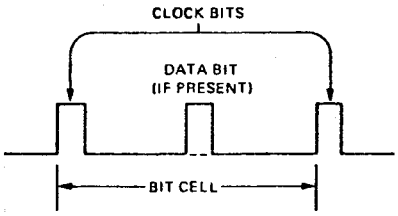


FIGURE 1-6
BIT CELL

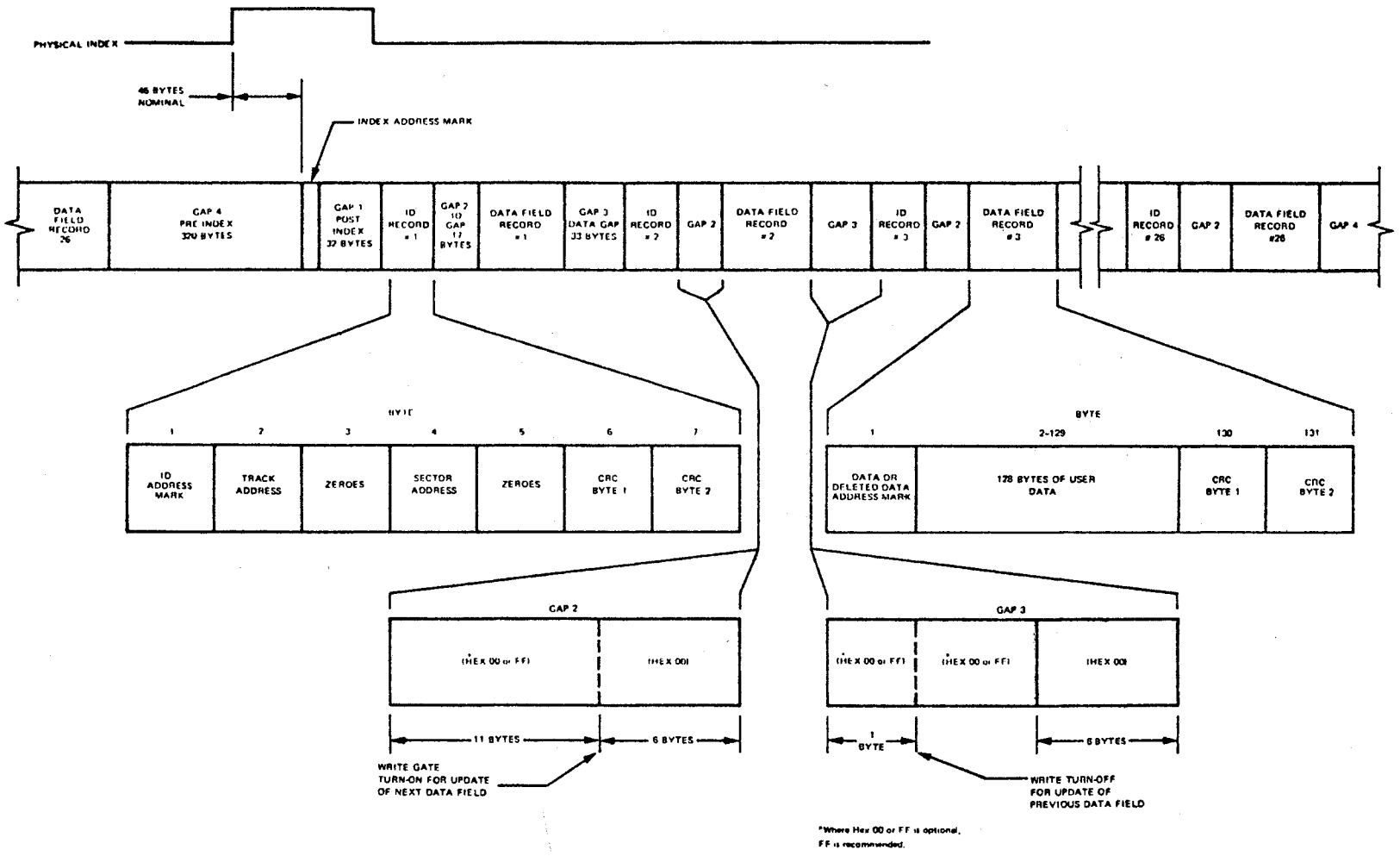


FIGURE 1-7

TRACK FORMAT

- Gap 3 - Data Gap
  The thirty-three bytes between the Data field and the next ID field are defined as Gap 3 (Data Gap). As with the ID Gap, the Data Gap may vary slightly in length after the adjacent Data field has been updated.

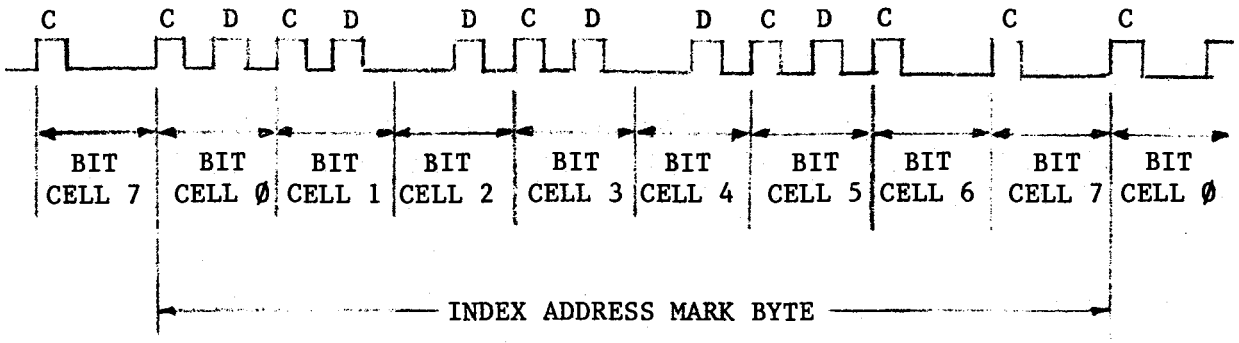
- Gap 4 - Pre-Index Gap
  The three hundred and twenty bytes between the last Data field on a track and the Index Address Mark are defined as Gap 4 (Pre-Index Gap). Initially, this gap is nominally 320 bytes in length; however, due to write frequency tolerances and diskette speed tolerances this gap may vary slightly in length. Also, after the data field of record 26 has been updated this gap may again change slightly in length.

Address Marks:

Address Marks are unique bit patterns one byte in length which are used to identify the beginning of ID and Data fields and to synchronize the deserializing circuitry with the first byte of each field. Address Mark bytes are unique from all other data bytes in that certain bit cells do not contain a clock bit (all other data bytes have clock bits in every bit cell.) There are four different types of Address Marks used. Each of these is used to identify different types of fields.

- Index Address Mark
  The Index Address Mark is located at the beginning of each track and is a fixed number of bytes in front of the first record. The bit configuration for the Index Address Mark is shown in Figure 1-8.

- ID Address Mark
  The ID Address Mark byte is located at the beginning of each ID field on the diskette. The bit configuration for this Address Mark is shown in Figure 1-9.



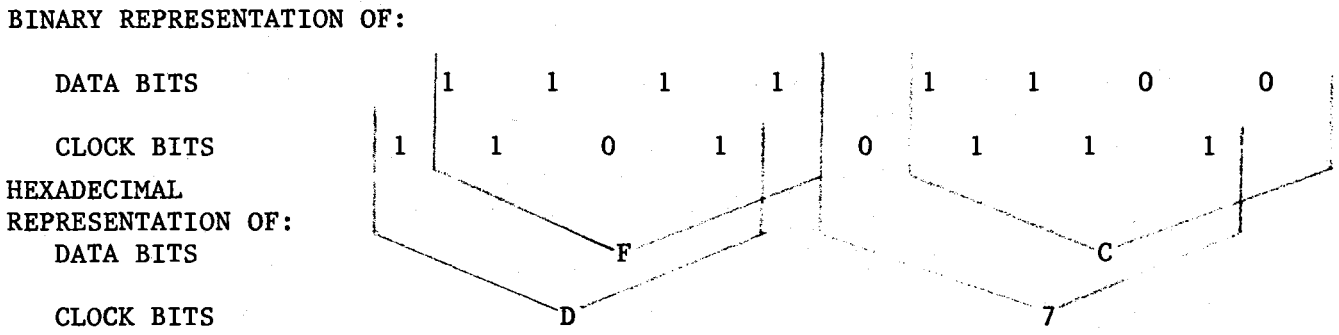
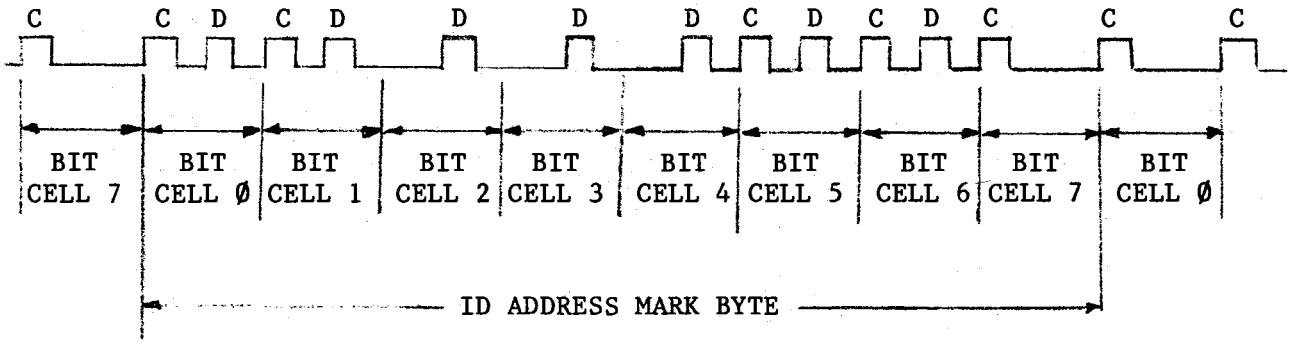


FIGURE 1-8

INDEX ADDRESS MARK



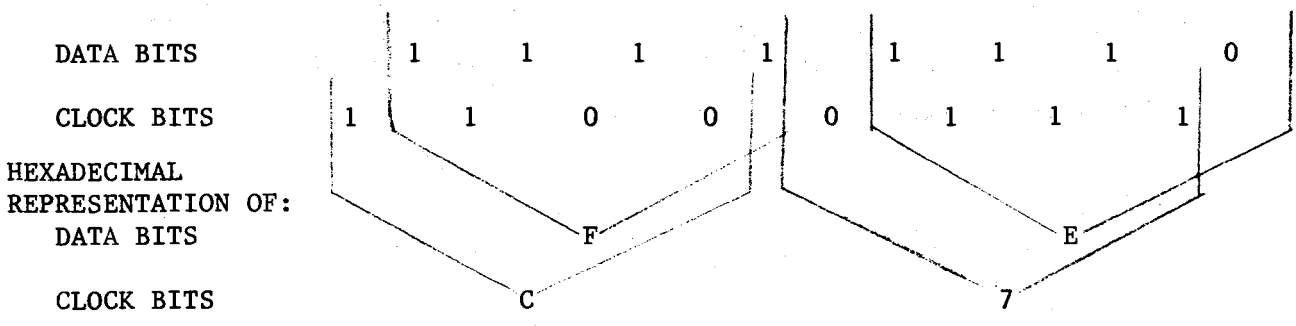


FIGURE 1-9

ID ADDRESS MARK

- Data Address Mark

  The Data Address Mark byte is located at the beginning of each Data Field on the diskette. The bit configuration for this Address Mark is shown in Figure 1-10.

- Deleted Data Address Mark

  The Deleted Data Address Mark byte is located at the beginning of each "deleted Data" Field on the diskette. The bit configuration for this Address Mark is shown in Figure 1-11. (Note: This bit pattern may be used to flag a data sector as unused or unusable.)

CRC Bytes:

Each field written on the diskette is appended with two Cyclic Redundancy Check (CRC) bytes. These two CRC bytes are generated from a cyclic permutation of the data bits starting with bit zero of the address mark and ending with bit seven of the last byte within a field (excluding the CRC bytes). This cyclic permutation is the remainder from the division of the data bits in the field (represented as an algebraic polynomial) by a generator polynomial G(X). For all fields recorded on a diskette, this generator polynomial is:

$$G(X) = X^{16} + X^{12} + X^{5} + 1$$

When a field is read back from a diskette, the data bits (from bit zero of the address mark to bit seven of the second CRC byte) are divided by the same generator polynomial G(X) and a non-zero remainder indicates an error within the data read back from the drive while a remainder of zero indicates the data has been read back correctly from the diskette or an undetectable error has been read back.

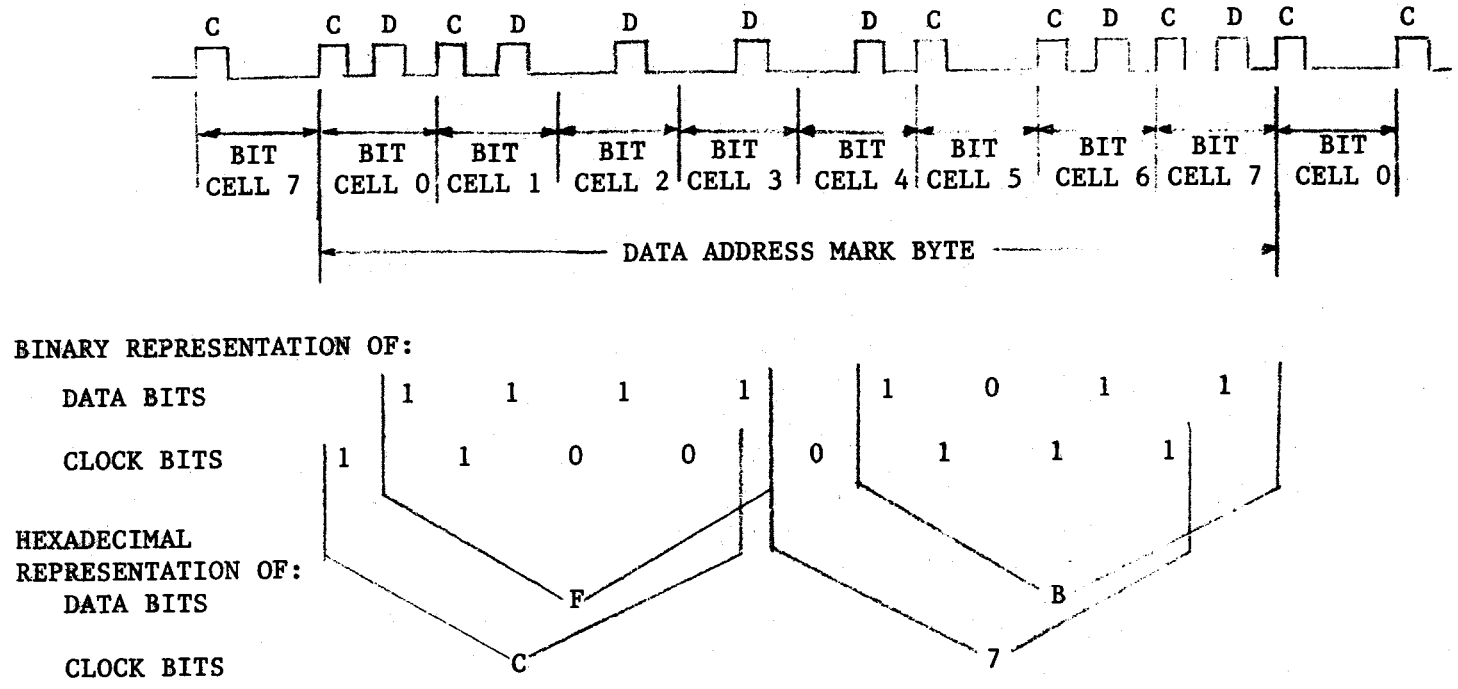


FIGURE 1-10

DATA ADDRESS MARK

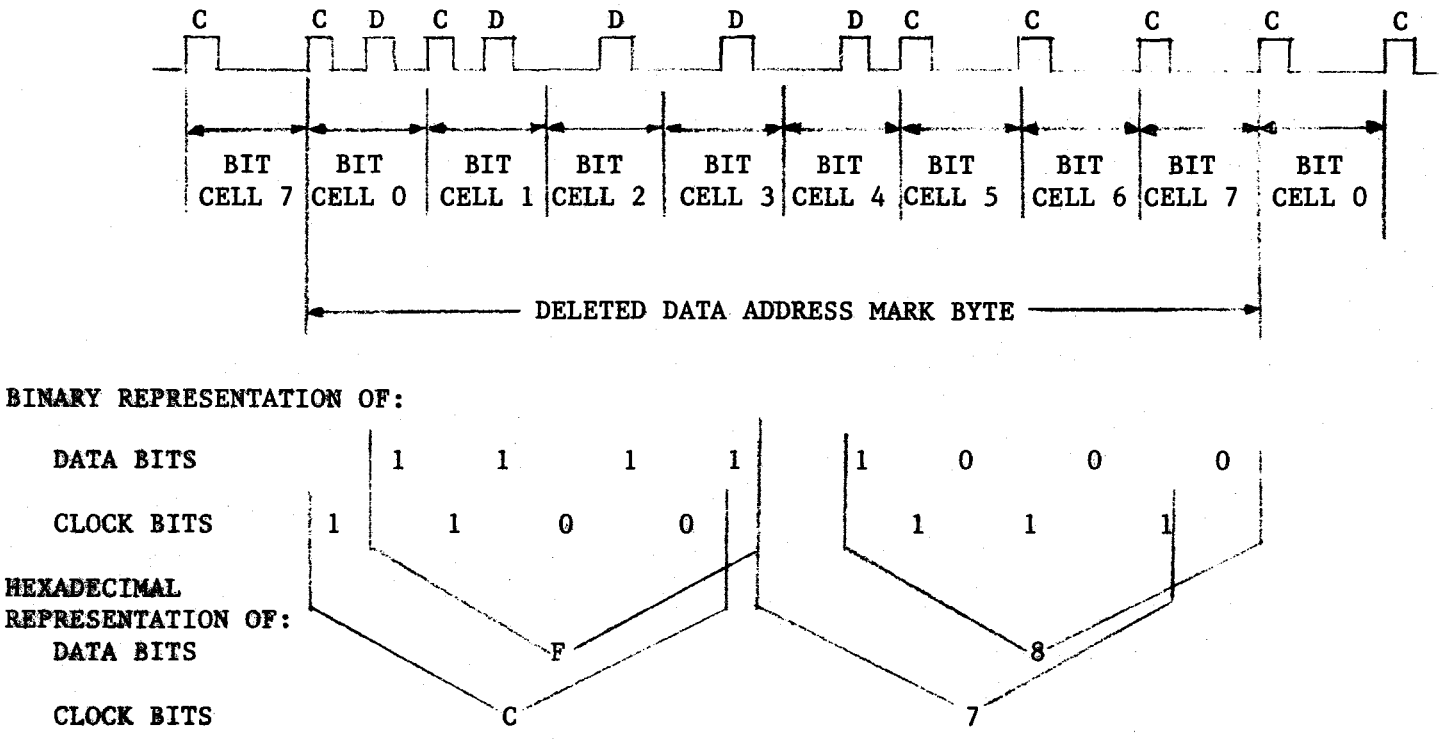


FIGURE 1-11

DELETED DATA ADDRESS MARK

CHAPTER 2
OPERATIONAL SUMMARY AND PROGRAMMING CONSIDERATIONS

All diskette operations are initiated by a Central Processor Unit (CPU) within the Computer system. Once initiated, however, the Diskette Channel completes the specified operation without further intervention on the part of the CPU. From the CPU's point of view, there are only three general steps required to complete any diskette operation:

- The CPU must prepare and store in system memory an I/O Parameter Block (IOPB) for each operation to be performed. If multiple operations are desired, several IOPBs may be linked together in the proper order. An IOPB (ten bytes) specifies a particular diskette operation and provides all of the parameters required for execution of that operation.

- The CPU must then pass the memory address of the first (or only) IOPB to the Diskette Channel.

- The CPU must process the result information from the Diskette Channel upon completion of the operation(s).

The preparation of the IOPB(s) by the CPU, in itself, requires no interaction with the Diskette Channel. The passing of the memory address for the IOPB and the result processing, however, do require interaction. Seven channel commands have been defined to allow the CPU to perform these interactive steps. Four of the channel commands are the result of the CPU executing an output instruction to a dedicated I/O port address, while the other three commands are the result of input instructions to dedicated ports. The seven channel commands are:

1) Write memory address lower (output)

2) Write memory address upper and start the diskette operation (output)

3) Stop diskette operation (output)

4) Reset the channel (output)

5) Read subsystem status (input)

6) Read result type (input)

7) Read result byte (input)

The CPU outputs the memory address of the first (or only) IOPB by executing channel commands 1 and 2. Upon execution of channel command 2, the Diskette Channel will request master control of the SBC system bus, fetch the diskette instruction and associated parameters from the IOPB(s), and proceed to perform the specified diskette operation(s). The diskette instruction byte in the IOPB can specify any one of seven diskette operations:

1) Recalibrate (seek track 00)
2) Seek
3) Format a track
4) Write data (with data address marks)
5) Write data (with deleted address marks)
6) Read data
7) Verify CRC

The Diskette Channel can interrupt the CPU when the operation (or group of linked operations) is completed or when the drive's ready status changes. The host system software can implement its CPU interrupt mechanism via this direct interrupt feature or it can

"poll" the Diskette Channel by executing channel command 5 (read subsystem status). When the CPU determines that the operation sequence has been completed (either by receiving an interrupt request or by reading the interrupt status), the CPU should execute channel commands 6 and 7 (read result type and read result byte) to determine whether the diskette operations were successfully completed, and if not which type of error occurred.

Thus, in summary, we see that certain channel commands are executed by the CPU to point the Diskette Channel to an IOPB (or group of linked IOPB's) in system memory, and initiate the operation sequence. The Diskette Channel, then, accesses the IOPB(s) to perform the diskette operation(s) specified by the instruction byte of the IOPB(s). The Diskette Channel will, if enabled by the IOPB, generate an I/O complete interrupt request:

- upon completion of each unchained diskette operation,
- upon completion of the last operation in a chain of linked diskette operations, or
- upon detection of an error during an intermediate operation in a chain of linked operations.

The CPU, then, executes other channel commands to determine the result(s) of the diskette operation(s).

In the preceding paragraphs, we have mentioned the channel commands, diskette operations and the IOPB without defining them explicitly. That is because up until now, our primary intention has been to identify clearly the function of each in the overall operation of the Diskette Channel. In the subsequent sections of this chapter, however, we will provide detailed information on the use and format of the channel commands (Section 2.1), the diskette operations (Section 2.2) and the IOPB (Section 2.3). Section 2.4 will define each of the error conditions that can be indicated when the "read result byte" channel command is executed by the CPU.

2.1 CHANNEL COMMANDS

There are seven channel commands to which the Diskette Channel will respond. Four of the channel commands are issued when a CPU in the computer system executes output (I/O write) instructions with the appropriate eight-bit I/O addresses. The other three commands are issued when the CPU executes input (I/O read) instructions with the appropriate I/O addresses.

When the CPU executes one of the output channel commands, it activates the I/O write (IOWC/) line and duplicates the appropriate 8-bit I/O address on address lines ADR0/ - ADR7/ and ADR8/ - ADRF/ of the SBC system bus. Depending on the particular channel command, the CPU may also place relevant data on data lines DAT0/ - DAT7/ of the SBC bus. The CPU maintains the data lines until the Diskette Channel returns the transfer acknowledge (XACK/) signal.

When the CPU executes one of the input channel commands, it activates the I/O read (IORC/) line and duplicates the appropriate I/O address on both halves of the SBC bus. The CPU expects the Diskette Channel to activate the transfer acknowledge (XACK/) line when it has placed the requested data on data lines DAT0/ - DAT7/.

The Diskette Channel differentiates between the different channel commands by interrogating the I/O read (IORC/) and I/O write (IOWC/) lines and the three least significant address lines (ADR0/ - ADR2/). The five most significant I/O address lines (ADR3/ - ADR7) define the switch-selectable BASE address for the Diskette Channel.

If the Diskette Channel is not busy, it will respond to an output channel command within 3 microseconds. If it is busy, the "write MA lower" and "write MA upper" commands are ignored; no acknowledge is returned. [IMPORTANT NOTE: Because no acknowledge is returned in this case, it could be possible to "hang up" the host system if the system does not include a Fail Safe time-out provision.] The "stop"

and "reset" commands, however, are acknowledged even if the Diskette Channel is busy. "Stop" is stored and executed at the end of the current diskette operation. "Reset" is executed immediately (if issued during a data write operation, garbled data will be written).

The Diskette Controller responds to "read subsystem status" and "read result type" input channel commands within 1 microsecond. The information returned in response to a "read subsystem status" command is always valid. The eight bits of data returned in response to a "read result type" command, however, are only valid if the Diskette Channel had previously issued an interrupt request to the CPU. The Diskette Channel will, if not busy, respond to a "read result byte" input command within 3 microseconds. If the Diskette Channel is busy, however, it ignores the "read result byte" command (i.e., no acknowledge is returned). The "read result type" and "read result byte" commands must be executed sequentially ("read result type" first), and should be executed only in response to an interrupt request from the Diskette Channel; execution at other times could produce erroneous result data.

The use and format of each of the seven channel commands is described below: (Note: In the following discussion, "BASE" refers to the switch selectable address on the Channel Board.)

Write memory address lower (output)

This channel command outputs the low order byte of the 16-bit memory address that points to byte 1 ("channel word") of the first (or only) IOPB.

System address bus: BASE + 1

System data bus: Eight least significant bits of the 16-bit memory address that point to the first IOPB.

Write memory address upper and start the diskette operation (output)

This channel command outputs the high order byte of the 16-bit memory address that points to byte 1 of the first (or only) IOPB. This command also causes the Diskette Channel to begin executing the diskette operation specified in byte 2 (instruction byte) of

the addressed IOPB.

System address bus: BASE + 2

System data bus: Eight most significant bits of the 16-bit memory address that point to the first IOPB.

Stop diskette operation (output)

This output channel command causes the Diskette Channel to cease operations after completing the current diskette operation. The Diskette Channel will not proceed to the next linked IOPB.

System address bus: BASE + 3

System data bus: Not used.

Reset Diskette System (output)

This output channel command causes all control logic in the Diskette Channel to be reset to an initialized state. If this command is issued while a "write data" diskette operation is in progress, the data in the sector currently being written will be garbled. This command is intended to clear a "hang up" in the Diskette Channel.

System address bus: BASE + 7

System data bus: Not used.

Read subsystem status (input)

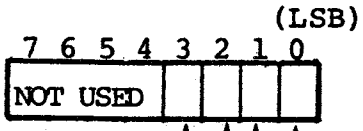
This input channel command causes the Diskette Channel to return four status bits to the CPU. The four bits are:

- bit 0 - ready status of drive Ø.
- bit 1 - ready status of drive 1.
- bit 2 - state of the channel's interrupt flip-flop
- bit 3 - controller presence indicator

These indications allow the user's system to monitor the operation of the Diskette Channel.

System address bus: BASE + ø

System data bus:



logical 1=controller present
logical 0=controller not present

logical 1=interrupt pending
logical 0=no interrupt pending

logical 1=drive 1 ready
logical 0=drive 1 not ready

logical 1=drive 0 ready
logical 0=drive 0 not ready

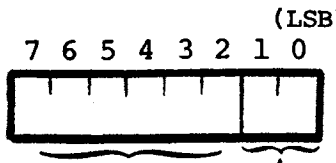
Read result type (input)

This input channel command causes the Diskette Channel to return eight bits of information to the CPU. The two least significant bits specify one of four different types of result byte (see next paragraph) associated with diskette operations. The remaining six bits are interpreted according to the code presented in the two least significant bits.

Note: In normal operation, a read result type command must be issued to clear the interrupt indicator. In case of a hardware malfunction, control reset ("master clear") can also be used to clear the indicator.

System address bus: BASE + 1

System data bus:



*Block number if type code = 01; otherwise, these bits are not used.

Type Code
- 00 - I/O Complete (unlinked); result byte contains error bits.
- 01 - I/O complete (linked); result byte contains error bits; six most significant bits of this word are block number.
- 10 - Result byte contains diskette ready status.
- 11 - Reserved

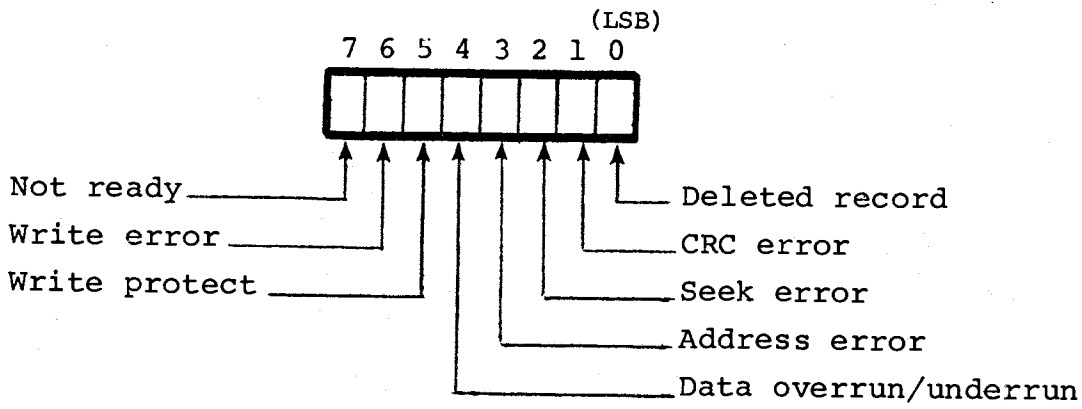
*NOTE: A block number is a 6-bit binary number that uniquely identifies an IOPB (See Section 2.3). The block number is returned in the result type word to identify the IOPB associated with the current interrupt. This scheme is only required when several IOPB's are linked together to perform multiple operations, because there is no uncertainty about the origin of an interrupt request when only a single IOPB exists.

Read result byte (input)

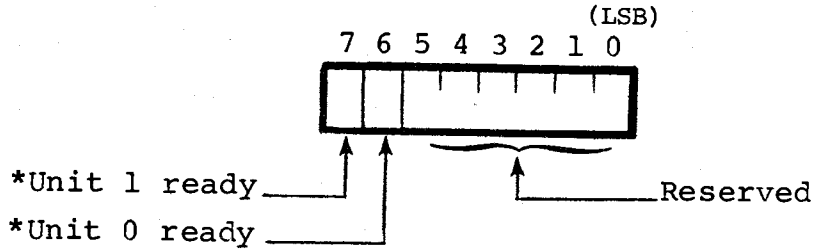
This input channel command causes the Diskette Channel to return eight bits of information to the CPU. The interpretation of these bits is dependent upon the type code returned in the result type word (see previous paragraph). The "read result byte" channel command should only be executed after a "read result type" command has been executed.

System address bus: BASE + 3

System data bus: If the type code in the result type word = 00 or 01, the result byte, input on the data bus, will contain error bits (see Section 2.4 for error explanations) and will be formatted as follows:



If the type code = 10, the controller has detected a change in the ready status of a drive and the contents of the result byte will indicate the current ready status of the diskette drives:



*NOTE: A logical 1 means that the drive is currently ready; a logical 0 means the drive is not ready. It is the responsibility of the host system software to maintain appropriate tables to track these status changes. There is one instance in which a drive can appear "not ready" to the host system, when in fact it is ready. For example, assume that while drive 0 is selected, drive 1 just goes not ready then returns to the ready state (perhaps the diskette platter was changed). When the drive 0 operation is completed, the diskette controller will return two consecutive status change interrupts, the first showing drive 1 not ready, the second showing drive 1 ready. The first interrupt, indicating drive 1 to be not ready, is returned even though the drive is now actually ready because it is important that the operator know that the ready status of the drive changed while the other drive was selected. For instance, this would protect against inadvertently accessing an "unknown" disk, if the drive went not ready then ready again because someone changed diskette platters.

2.2 DISKETTE OPERATIONS

The Diskette System is capable of performing seven different operations: recalibrate, seek, format track, write data (with data marks), write data (with deleted data marks), read data, and verify CRC. To initiate any diskette operation (or group of linked operations), the CPU will output both bytes of the 16-bit memory address that points to the first byte of an I/O Parameter Block (IOPB). The second byte in the IOPB specifies one of the seven diskette operations (see Section 2.3 for IOPB format). After the Diskette System receives the upper byte of the 16-bit memory address, it accesses the IOPB to determine the operation to be performed and to acquire the various parameters that are necessary for execution of the diskette instruction. The Diskette System will perform the specified operation, then set its interrupt flip-flop. If several IOPB's are linked together (see Section 2.3), the Diskette System will perform all of the specified operations (each IOPB specifies one diskette operation) before interrupting the CPU with a request for service.

NOTE: The Diskette Channel automatically unloads the read/write head after a fixed length of time following a diskette operation or group of linked operations. This feature is meant to reduce head wear. The feature is implemented by counting index pulses after a "read result byte" channel command is executed. When the specified count is achieved, the head is unloaded, and the count is re-initialized. At present, the count is set for 6; that is, the head will remain loaded for at least five complete revolutions following each diskette operation or group of linked diskette operations.

The seven diskette operations are defined in the following paragraphs: (Refer to pages 2-21 and 2-18 for the exact bit patterns required to execute each operation.)

RECALIBRATE

This operation causes the head of the selected diskette unit to be moved over track 00. The diskette drive's track 0 sensor is sampled to determine successful completion of this operation. This is often the first instruction executed after a diskette is loaded, or when a seek error occurs (see Section 2.4).

SEEK

This operation causes the head of the selected diskette unit to be moved to the track specified in byte 4 of the IOPB. The Diskette Channel will verify the head position by reading the track address from the diskette platter before completing the operation. If at the completion of the head movement, the head is not over the expected track, a "seek error" will be indicated (see Section 2.4).

FORMAT TRACK

This operation initializes the track specified in byte 4 of the IOPB, by writing all address marks, gaps, address fields and data fields, as shown in Figure 2-1. The various address marks and fields are defined in Section 1.2.

The method of assigning logical sector addresses, which are written into the sector address fields, is specified by bit 6 of the first IOPB byte (the channel word). If this bit is equal to logical 0 the sequence of logical sector addresses will match the physical sequence on the diskette (i.e., sector address "01" is written into the first physical sector after the index mark, sector address "02" is written into the second physical sector, and so on). In addition, the data byte stored in the memory location specified by the 16-bit buffer address contained in bytes 6 and 7 of the IOPB will be written into the 128-byte locations of each sector's data field. No other data bytes need to be stored in this buffer.

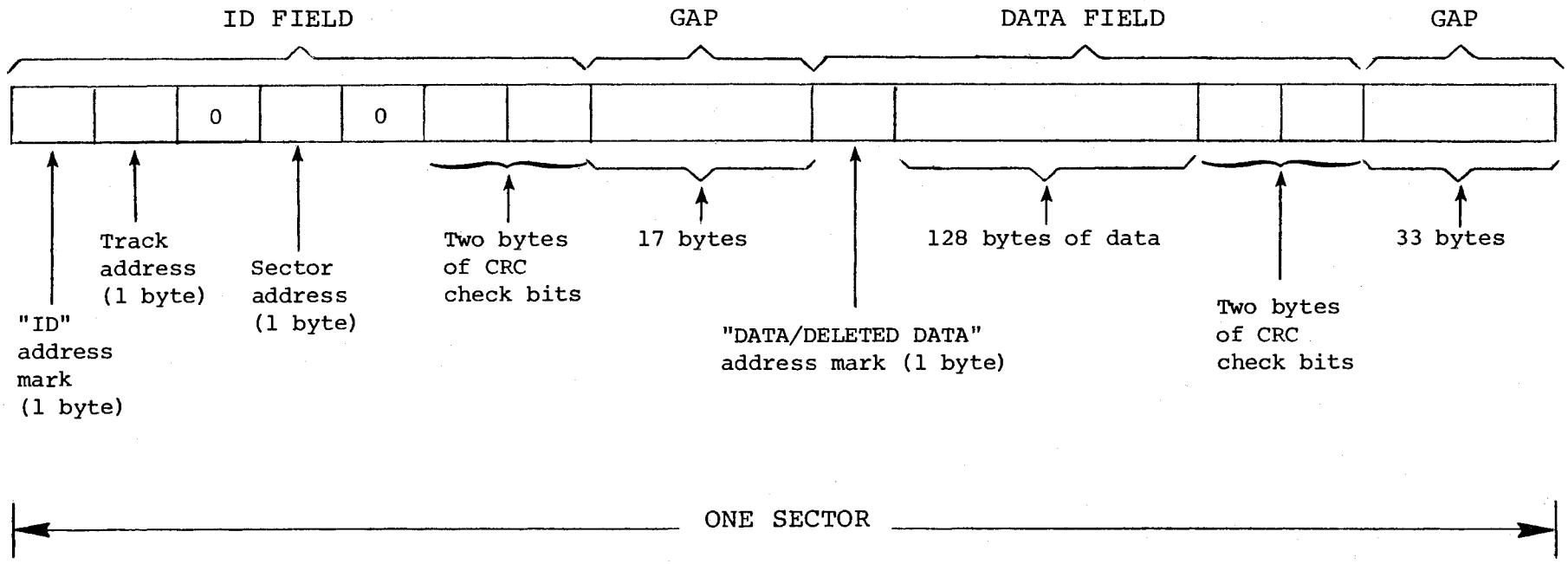


FIGURE 2-1

SECTOR FORMAT

If, on the other hand, the sequence of logical addresses being assigned to the sectors is "random" (that is, do not match the physical sequence of sectors), bit 6 of the channel word will be equal to logical 1, and 52 bytes (26 pairs) of data will be stored in memory beginning at the 16-bit buffer address contained in bytes 6 and 7 of the IOPB. Each of the 26 pairs of data bytes will specify the logical sector address to be written into the sector address field of the corresponding physical sector, and the data character which will be written (128 times) into the data field portion of that sector. For example, if the first four bytes of the buffer are:

Byte	Contents (hex)	
1	01	First sector number
2	FF	Data
3	0E	Second sector number
4	00	Data
.	.	
.	.	

Then, sector address "01" will be written into the sector address field of the first physical sector after the index mark, and "FF_{16}" (all ones) will be written into each of the 128 byte locations in the data field portion of this sector. The sector address "$0E_{16}$" (14_{10}) will be written into the sector address field of the second physical sector (i.e., the sector which is physically next to the first sector), and "00_{16}" (all zeros) will be written into each of the 128 byte locations in the data field portion of this sector. And so on, until a logical sector address has been written into the sector address field of each of the 26 physical sectors on the track, and a data byte is written into each of the 128 byte locations in the data field portion of each of the 26 sectors.

The firmware implementation of the format command is such that in order to format track n ($n \neq 0$), track n-1 must already be formatted (i.e., already have readable address information written into it). Track 0 can always be formatted, even if no valid address information is written on the disk.

During formatting, a "data mark" (i.e., a character which has a clock pattern equal to $C7_{16}$ and a data pattern equal to FB_{16}; see Figure 2-2) is written into the "data/deleted data address mark" character position of each sector (i.e., the character position immediately preceding the 128 byte data field).

If, when the format track operation is initiated, the head is not already positioned over the track specified in byte 4 of the IOPB, the format track instruction will cause the head to move (seek) to the proper track before the actual formatting begins.

WRITE DATA

This operation transfers N x 128 bytes of contiguous data from memory to the diskette. N represents the number of sectors to be written. N is specified by the contents of byte 3 of the IOPB. The 16-bit buffer address stored in bytes 6 and 7 of the IOPB specifies the memory location containing the first data byte to be transferred. The contents of bytes 4 and 5 of the IOPB (track and sector addresses, respectively) specify the logical address of the first sector to be written into.

Each 128 byte data field will be preceded by a "data" address mark (see Figure 2-2) that is used for synchronization. Two bytes (16 bits) of CRC check bits will be generated and written after each data field; the CRC bytes are generated from the address mark, as well as the 128 data bytes.

A multi-sector operation (i.e., $N \geq 2$) may begin at any sector, but must not go beyond the last logical sector on a track (sector 26).

If the head is not already positioned over the track specified in byte 4 of the IOPB, the write data instruction will cause the head to move (seek) to the proper track before the actual writing begins.

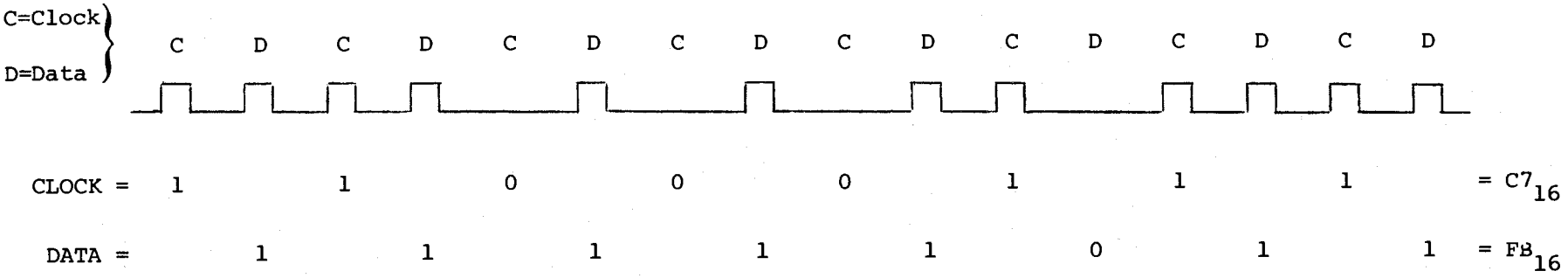


FIGURE 2-2
"DATA" ADDRESS MARK

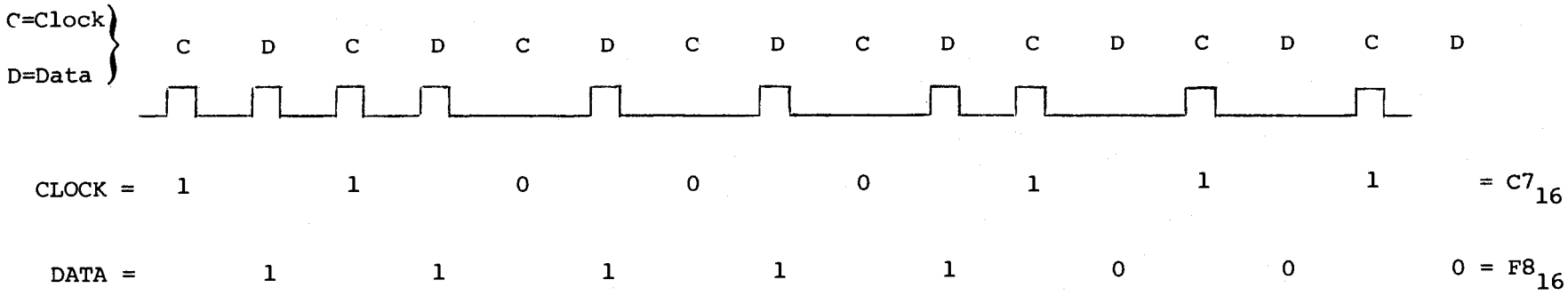


FIGURE 2-3
"DELETED DATA" ADDRESS MARK

WRITE "DELETED" DATA

This operation is identical to the WRITE DATA operation, described above, except that each 128 byte data field is preceded by a "deleted data" address mark, shown in Figure 2-3.

READ DATA

This operation transfers N sectors of data (128 bytes per sector) from diskette to memory. N is specified by the contents of byte 3 of the IOPB. The contents of bytes 4 and 5 of the IOPB (track and sector addresses, respectively) specify the logical address of the first sector to be read. The 16-bit buffer address stored in bytes 6 and 7 of the IOPB specifies the memory location into which the first data byte will be written.

Two bytes of CRC check bits will be generated as each sector is being read. When the "data" address mark and all 128 data bytes of a sector have been read, the generated CRC bits are compared with the 16 CRC bits previously written. If there is a mismatch, a CRC error is indicated (see Section 2.4).

A multi-sector operation (i.e, N$\geq$2) may begin at any sector, but must not go beyond the last logical sector on a track (sector 26).

If the head is not already positioned over the track specified in byte 4 of the IOPB, the read data instruction will cause the head to move (seek) to the proper track before the actual data reading begins.

VERIFY CRC

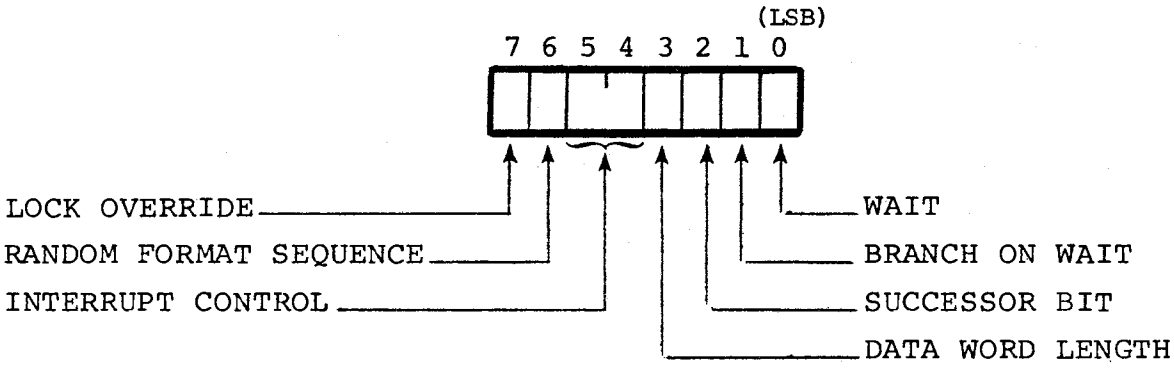
This operation is identical to the READ DATA operation, described above, except that no data is transferred to memory.

2.3 I/O PARAMETER BLOCK

The CPU in the SBC system initiates a diskette operation, or group of linked operations, by outputting a 16-bit address that points to the beginning (the channel word) of the first (or only) I/O Parameter Block (IOPB) in system memory. The Diskette Channel then accesses the IOPB. An IOPB specifies one of the seven diskette operations (see Section 2.2) and provides all of the parameters required for the completion of that operation. An IOPB consists of ten bytes, as shown in Figure 2-4.

Byte 1. Channel Word

This byte contains channel control information to be used by the Diskette System. Bit assignments in this byte are as follows:



The "random format sequence" bit (6) specifies the method of assigning logical sector addresses when formatting a track. If this bit is reset (logical 0), sector addresses are assigned in sequential order. If this bit is set (logical 1), sector addresses are assigned in random order according to the pattern listed in the 52 byte memory buffer, which begins at the location addressed by the contents of IOPB bytes 6 and 7. (Refer to the description of the FORMAT TRACK operation in Section 2.2.)

The "interrupt control" bits (4 and 5) enable or disable Diskette Channel interrupts according to the scheme shown in Table 2-1.

BYTE	
*1	Channel Word
2	Diskette Instruction
3	Number of Records
4	Track Address
5	Sector Address
6	Buffer Address (Lower)
7	Buffer Address (Upper)
8	Block Number
9	Next IOPB Address (Lower)
10	Next IOPB Address (Upper)

* The 16-bit address output to the Diskette System by the two "Write MA" channel commands points to the first byte of an IOPB.

FIGURE 2-4
I/O PARAMETER BLOCK (IOPB) FORMAT

TABLE 2-1

INTERRUPT CONTROL BITS

BIT: 5	4	FUNCTION
0	0	I/O complete interrupt request to be issued (a) upon completion of an unchained diskette operation, (b) upon completion of the last operation in a chain of linked operations, or (c) upon detection of an error in any intermediate operation in a chain of linked operations.
0	1	All I/O complete interrupts are disabled.
1	0	I/O complete interrupt to be issued after current operation even though it is an intermediate link in a chain.
1	1	Illegal code

NOTE: The interrupt control bits do not affect interrupt requests which are issued as the result of a change in diskette ready status.

The "data word length" bit (3) must be reset (logical 0) when the Diskette Channel is being used with 8-bit systems, or set (logical 1) when being used with 16-bit systems. This bit must be logical 0 when being used with the SBC system (an 8-bit system).

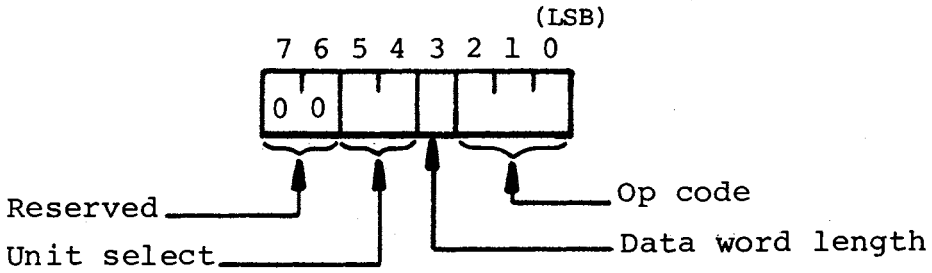
The "successor" bit (2) will be reset (logical 0) if the current IOPB is the last (or only) IOPB to be executed. This bit will be set (logical 1) if at least one more IOPB is to follow. If this bit is reset, an I/O complete interrupt request will be issued when the operation specified by the current IOPB is completed. If there is a successor IOPB, that IOPB will begin at the memory location specified by bytes 9 and 10 of the current IOPB.

The "branch on wait" (1) and "wait" (0) bits are interrelated. If the "wait" bit is reset (logical 0), the Diskette Channel immediately performs the diskette operation specified by the current IOPB. If the "wait" bit is set (logical 1), however, the Diskette Channel examines the "branch on wait" bit. If "branch on wait" is set (logical 1) and "wait" is set, the Diskette Channel performs an unconditional branch to the memory location specified by the 16-bit address in bytes 9 and 10 of the current IOPB; the next IOPB to be executed should reside at this memory location. If "branch on wait" is reset (logical 0) but "wait" is set, the Diskette Channel will idle for 10 msec., then examine the "wait" bit again, remaining in this loop until "wait" is reset. This gives the programmer greater flexibility in establishing a sequence of operations to be performed by the Diskette Channel. Note that "branch on wait" <u>MUST</u> be reset (logical 0) if "wait" is reset (logical 0). The Diskette Channel sets the "wait" bit after completing the operation specified in the IOPB (unless the "lock override" bit is set). This protects again inadvertent execution of an uninitialized IOPB.

The "lock override" bit (7) when set, specifies that the "wait" bit is <u>not</u> to be set upon completion of the operation specified in the IOPB. This prevents the IOPB from being overwritten by the controller; this is a useful feature during system debugging.

Byte 2. Diskette Instruction

This byte specifies the diskette operation to be performed and identifies the diskette unit to be used:



The "unit select" bits (4-5) specify drive #0 when reset (logical 00_2), or drive #1 when set (logical 11_2).

The "data word length" must contain the same value as the corresponding bit in the channel word (byte 1).

The "op code" bits (0-2) specify one of the seven diskette operations (refer to Section 2.2):

BIT: 3	2	1	OPERATION
0	0	0	No operation
0	0	1	Seek
0	1	0	Format track
0	1	1	Recalibrate
1	0	0	Read data
1	0	1	Verify CRC
1	1	0	Write data
1	1	1	Write "Deleted" Data

Byte 3. Number of Records

This binary number specifies the number of sectors to be transferred. Multi-sector operations are allowed, but they must not go beyond the last sector on a track (sector 26); that is, an address error (see Section 2.4) will be indicated if (starting sector address) + (number of records)$>26_{10}$. Therefore, the maximum block transfer is 26 sectors (from sector 1 to sector 26).

Byte 4. Track Address

This binary number identifies the track. Acceptable values are 0 to $4C_{16}$ (76_{10}), inclusive.

Byte 5. Sector Address

Bits 4 through 0 of this byte contain a binary number which specifies the first sector to be accessed during transfer operations. Acceptable values are 1 to $1A_{16}$ (26_{10}), inclusive. Bit 5 of this word MUST correspond to bit 5 (select bit) of the diskette instruction word (byte 2). Bits 6 and 7 are not used.

Byte 6. Buffer Address (lower)

This byte contains the eight least significant bits of the 16-bit buffer memory address.

Byte 7. Buffer Address (upper)

This byte contains the eight most significant bits of the 16-bit buffer memory address. Bytes 6 and 7 together contain the 16-bit address of the first word of the buffer in system memory. During read data operations, the data from the diskette is transferred to the buffer. During write operations, data from the buffer is written to diskette. During format track operations, the address assignment pattern and/or the data field "format characters" are stored in the buffer. (Note: Bytes 6 and 7 must contain a valid address for all diskette operations.)

Byte 8. Block Number

This byte contains a 6-bit (right-justified) binary number that uniquely identifies the current IOPB. The block number allows the CPU to associate an I/O complete interrupt request from an intermediate link in a chain of IOPB's with the IOPB which actually caused the interrupt. The block number need only be initialized for linked IOPB's, since there can be no uncertainty when only a single IOPB exists.

Byte 9. Next IOPB Address (lower)

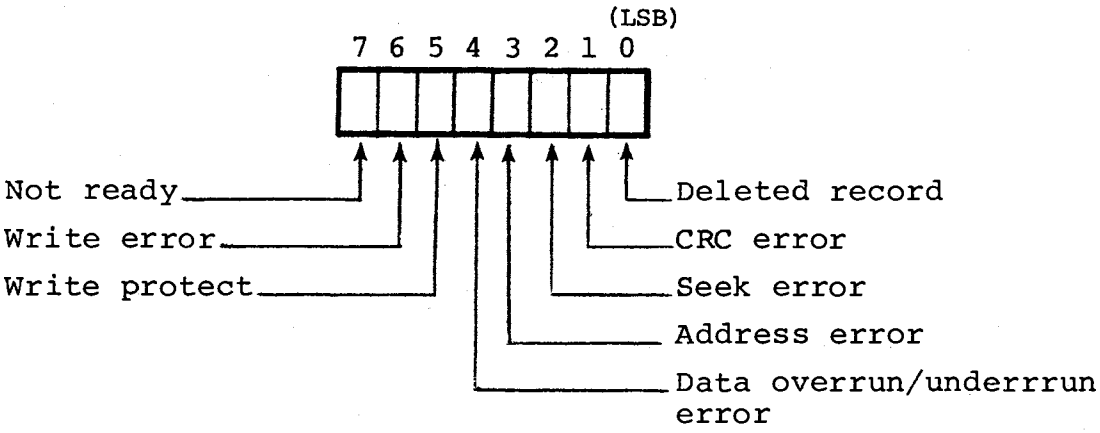
This byte contains the eight least significant bits of the 16-bit memory address that points to the beginning of the next IOPB in the chain.

Byte 10. Next IOPB Address (upper)

This byte contains the eight most significant bits of that 16-bit memory address that points to the first byte of the next IOPB in a chain. The Diskette System will access the next IOPB after the current operation if the "successor" bit (byte 1) in the current IOPB is set, or immediately if both the "wait" and "branch on wait" bits (byte 1) in the current IOPB are set.

2.4 ERROR INDICATIONS

If the CPU executes a "read result byte" channel command (in response to a "read result type" channel command which returned a code of 00 or 01), the Diskette Channel will return the following result word on the system data bus:



The bits are defined as follows:

Not ready. This bit (7) indicates that the selected unit was not ready or that the selected unit changed to a not ready state during an operation.

Write error. This bit (6) indicates that, during a write operation, a condition existed which precluded data integrity. This error is detected by the drive and monitored by the Diskette Channel controller. An example of a condition that could cause this error is an attempt to write through an unloaded head.

Write protect. This bit (5) indicates that the selected drive contains a diskette platter which is in the "read only" mode. This condition is checked on format track, write data (with data address marks) and write data (with deleted data address marks) operations.

Data overrun/underrun error. This bit (4) indicates that the Diskette System controller was not able to service a byte transfer request from the drive before the next request occurred. The data byte is "lost".

Address error. This bit (3) indicates that the disk address received from the CPU is invalid; that is:

- track address > 76_{10},
- sector address = 00,
- 32_{10} > sector address > 26_{10}
- sector address + number of records > 26_{10}

Seek error. This bit (2) indicates that, at the completion of a head movement sequence, the head is not positioned over the expected track. This bit indicates the Diskette System controller and/or drive are malfunctioning, and a recalibrate diskette operation (see Section 2.2) should be performed. Because all of the diskette operations may implicitly cause the head to move, a seek error can occur during any diskette operation.

CRC error. This bit (1) indicates that the two CRC characters generated during a read data or verify CRC operation were not the same as the two CRC characters appended to the data field (see Section 1.2) when it was written on diskette.

Deleted record. This bit (0) indicates that a sector addressed during a read data or verify CRC operation was preceded by a deleted data address mark.

Four other error conditions are indicated when more than one error bit is true:

ID CRC error. If the address error (3) and CRC error (1) bits are true, it indicates that the CRC characters generated during the reading of an ID field (see Section 1.2) were not the same as the CRC characters appended to the field when it was written by a format track operation.

SYNC error. If the deleted record (0) and CRC error (1) bits are true, it indicates that an unexpected address mark pattern (see Section 1.2) was encountered while trying to establish data synchronization. This usually indicates that the ID field is garbled.

NO ADDRESS MARK. If the address error (3), seek error (2) and CRC error (1) bits are true, it indicates that no address mark (see Section 1.2) was encountered for a full revolution of the diskette. This usually indicates that the track has not been formatted.

DATA MARK error. If the address error (3), seek error (2), CRC error (1), and deleted record (0) bits are true, it indicates that the data field of a particular sector was not preceded by either a data mark or a deleted data mark.

CHAPTER 3
THE CHANNEL BOARD

The Channel Board is the primary control module within the Diskette System. The Channel Board receives, decodes and responds to channel commands from the Central Processing Unit (CPU) in the OEM computer system. The Channel Board can access common system memory to determine the particular diskette operations to be performed and to fetch the parameters required for the successful completion of the specified operations. The Channel Board also monitors subsystem status and error conditions, and organizes these indications into "result type" and "result byte" words that can be read by a CPU in the computer system.

The control functions of the Channel and Interface Boards are provided by an 8-bit microprogrammed processor, implemented with Intel's Series 3000 Bipolar Microcomputer Set. The 8-bit controller includes four 3002 Central Processing Elements (2-bit slice per CPE), a 3001 Microprogram Control Unit and 512 x 32 bits of 3604 programmable-read-only-memory (PROM) which stores the microprogram. The processing and control capabilities of the diskette controller are achieved by execution of the microprogram.

The Channel Board and the Interface Board may reside within System 80 backplane, SBC 604/614 cardcage, or a custom backplane. The Channel Board, together with the Interface Board, constitute what we refer to as the Diskette Controller.

3.1 FUNCTIONAL ORGANIZATION OF THE CHANNEL BOARD

For description purposes, the circuitry on the Channel Board can be divided into six functional blocks:

- Channel command block
- Micro control unit (MCU) block
- Microprogram memory block
- Central processing element (CPE) block
- Data/clock shift register (SR) block
- Data flow control block

as shown in Figure 3-1.

The channel command block is responsible for recognizing and decoding channel commands being executed by a CPU in the computer system. When the channel command block recognizes the switch-selectable BASE address of the Diskette Operating System on the SBC system address bus, it decodes the three least significant address bits (ADRO/ - ADR2) to determine which of the seven channel commands is being executed (see Section 2.1). The three address bits are also latched and made available to the MCU block, which is ultimately responsible for controlling the diskette controller's response to a channel command. The channel command block also includes the interrupt latch which stores the fact that an interrupt request has been issued to the CPU by the microprogram.

The micro control unit (MCU) block accepts and decodes the three address bits from the channel command block (ADRO/ - ADR2/) specifying a channel command or the three least significant data outputs from the CPE block (DO-D2) specifying one of the seven diskette operations. The two groups of 3 bits select one of the ten routines which implement the channel commands and I/O operations. Having determined the microprogram routine to be executed, the MCU block then generates and outputs the appropriate nine-bit memory address from the microprogram memory. The MCU continuously examines the two flag control

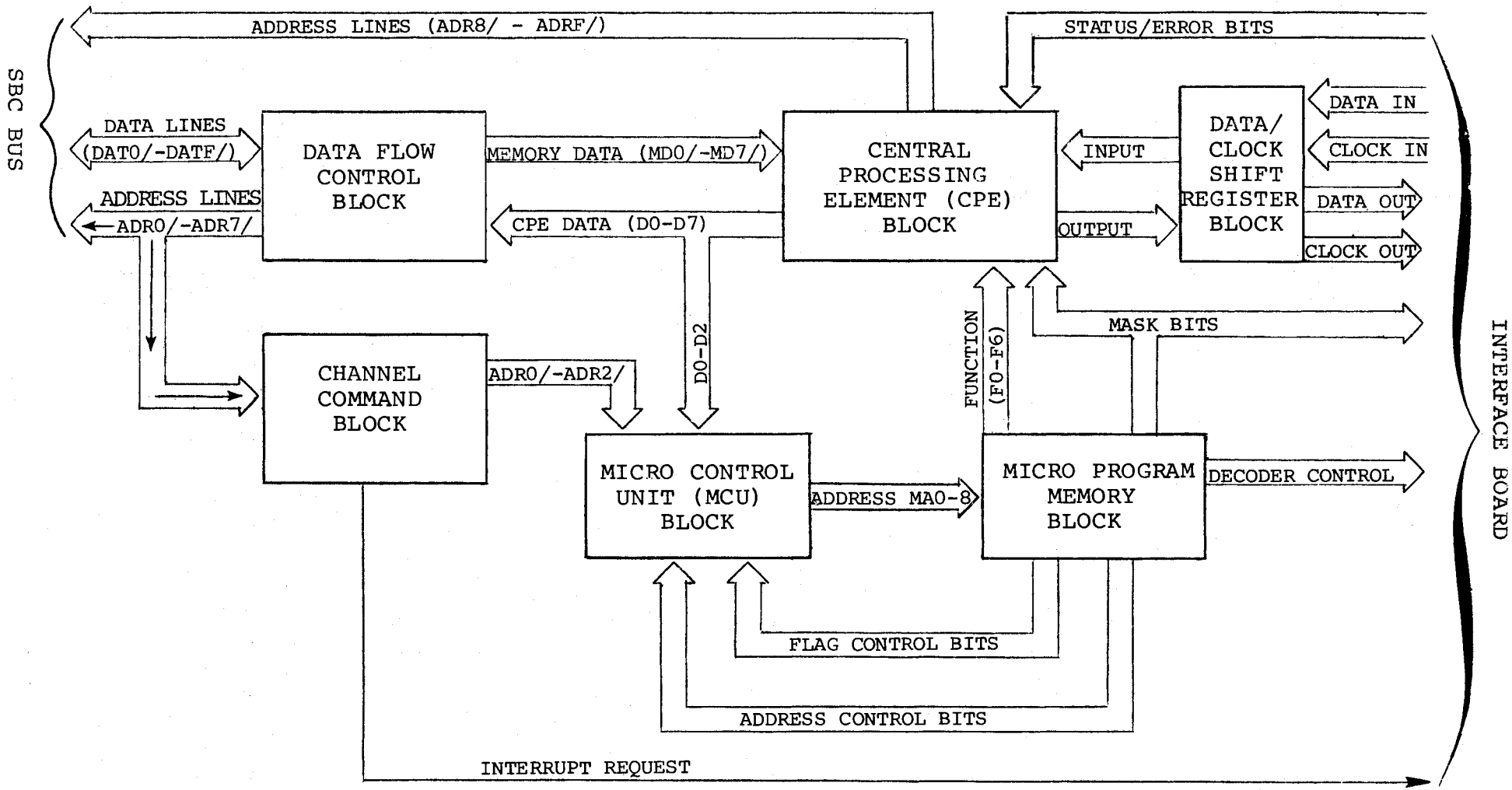


FIGURE 3-1

CHANNEL BOARD: FUNCTIONAL BLOCK DIAGRAM

lines and the seven address control lines (AC0 - AC6) from the microprogram memory block to determine the address of the next microinstruction to be fetched and executed.

The microprogram memory block, as its name implies, stores the microprogram. The microprogram memory is organized into 512 words of 32 bits each. The nine address bits from the MCU block determine which 32-bit microinstruction will be output from the microprogram memory. Nine bits of the microinstruction (the address control and flag control bits) are applied to the MCU block, as mentioned above, while the seven function bits (F0-F6) are applied to the CPE block and specify the operation to be performed by the processing elements. The other sixteen bits of the microinstruction words perform a variety of control functions, that will be described completely in Section 3.2.

The CPE block includes four Intel® 3002 Central Processing Elements, which form an 8-bit processor. The CPE block receives data from the data flow control block and the data/clock shift register block and receives status information from the Interface Board. The CPE can operate on these various types of input data under the direction of the function and mask control bits from the microprogram memory. The results of these arithmetic/logical operations can then be output onto the eight most significant system address lines (ADR8/ - ADRF/) or the eight CPE data lines (D0 - D7). D0 - D7 are, in turn, made available to the MCU block, the data/clock shift register block and the data flow control block.

The data/clock shift register block includes the shift registers that accept the serial data bits and the serial clock bits and input them, in parallel, to the CPE block during read operations. During write operations, the data and clock bytes are (parallel) loaded into the shift registers from the CPE block and shifted out (serially) to the Interface Board.

The data flow control block routes data from the CPE data lines (D0 - D7) to the eight least significant SBC system address lines (ADR0/ - ADR7/) or to either the lower or upper eight lines of the SBC system data bus (DATA0/ - DATA7/ or DATA8/ - DATAF/). This block also routes data from either half of the SBC data bus onto the memory data input lines (MD0/ - MD7/) that feed the CPE block.

3.2 THEORY OF OPERATION: CHANNEL BOARD

In this section we will describe the circuitry on the Channel Board. We will divide this theory of operation discussion into six sub-sections, each dealing with one of the functional blocks defined in Section 3.1.

The Channel Board accepts/transmits signals, data and power through three different PC edge connectors:

- P1 Bus connector (to/from system bus)
- P2 Controller connector (to/from Interface Board)
- J1 Test points only

To avoid any ambiguity when referring to connector pins in sub-sequent paragraphs, we will always list the connector as well as the pin whenever such references are required; for example, P1-57 refers to pin 57 on connector P1. Pin lists for the three connectors are provided in Section 3.3.

The schematic drawing (4 sheets) for the Channel Board is also provided in Section 3.3.

3.2.1 CHANNEL COMMAND BLOCK

The channel command block recognizes and decodes all channel commands directed to the diskette controller. This block includes an eight position switch (S1) for BASE address assignment, eight 74LS86 EXCLUSIVE-OR gates for address recognition, two 3205 three-to-eight decoders, two 7474 D-type flip-flops, a 74175 quad latch, and other assorted gating circuits, as shown on sheet 1 of the board schematic (Section 3.3).

Recall from Chapter 2, that the CPU specifies channel operations for the Diskette System by executing one of the seven channel commands. A channel command may be the result of either an input or output

instruction to a dedicated I/O port address on the Channel Board:

1) Write MA Lower (output to "BASE+1")
2) Write MA Upper and Start I/O (output to "BASE+2")
3) Stop Diskette Operation (output to "BASE+3")
4) Reset Channel (output to "BASE+7")
5) Read Subsystem Status (input to "BASE+0")
6) Read Result Type (input to "BASE+1")
7) Read Result Byte (input to "BASE+3")

The three least significant bits (ADR0/ - ADR2/) of the 8-bit I/O address (received at pins P1-51 through P1-58) differentiate between the various input or output channel commands. The five most significant address bits (ADR3/ - ADR7/) select the Channel Board if they match the BASE address that is assigned by setting five positions of switch S1. These five switch positions each feed one input on five EXCLUSIVE-OR gates. If ADR3/ - ADR7/ match the switch-selected BASE address, the 7410 NAND gate (A31-8) is activated and, in turn, enables one of the two 3205 decoders.

If an input channel command is being received, the RD CMD line (pin P2-60) will be true, and the 3205 decoder at A20 will be enabled. Address bits ADR0 - ADR2 (once inverted) are applied to the three data inputs on the 3205 section (A0 - A2), and activate one of three inverted outputs ($\overline{O_0}$, $\overline{O_1}$ or $\overline{O_3}$), depending on the channel command. If it is "read subsystem status" command, output 0 goes true and READ INT/ is asserted at pin P2-57. (On the Interface Board, READ INT/ is used to gate the device Ø and device 1 ready indicators onto system data bus lines Ø and 1, DATØ/ and DAT1/.) The low level on READ INT/ also enables two 8093 circuits, one of which transmits the output of the interrupt latch (INT/) to the data bit 2 line (DAT2/) of the system data bus. The other 8093 circuit transmits a low-level to the data bit 3 line (DAT3/), indicating that the diskette controller is present. INT/ is also passed to the Interface Board via pin P2-40.

If a "read result type" command is being received, the output from the decoder goes true, and the RD RI/ signal is generated (pin P2-37). The low level on RD RI/ pre-sets the interrupt latch (A37-10), thus removing the active-low system interrupt request (INT/).

The interrupt latch can subsequently be clocked reset (i.e., reset to the active-low state) by a pulse on the CLK line, when the central processing element block (Section 3.2.4) determines that an "I/O complete" or "ready status change" interrupt should be issued (also refer to Chapter 2).

If a "read result byte" command is being received, output 3 from the decoder will go true, and the 74175 quad latches are clocked, latching up address bits ADRO - ADR2. The three most significant quad latch outputs are made available to the micro control unit block (Section 3.2.2), which responds to this command via a routine stored in microprogram memory. Either read result command will generate the RD RES/ signal which is used by the data flow control block (Section 3.2.6) to gate the appropriate status word onto the system data bus.

If an output channel command is being received, the WRT CMD line (pin P2-53) will be true, and the other 3205 decoder will be enabled. Address bits ADRO - ADR2 are applied to inputs A0 - A2, causing one of the eight inverted decoder outputs to go true. If outputs 0, 1 or 2 go true, the WSUB1/ line (pin P2-48) is activated. If outputs 4, 5 or 6 from the decoder go true, the WSUB2/ line (pin P2-47) is activated. Either WSUB1/ or WSUB2/ will cause the 74175 quad latches to be clocked and latch up address bits ADRO - ADR2, just as a "read result byte" command did. The three most significant outputs of the quad latches are made available to the mecro control unit block which responds to the "write MA lower", the "write MA upper and start I/O" and the "read result byte" channel commands, using routines stored in microprogram memory. The response mechanisms for the other channel commands are implemented in hardware, not micro code.

If a "stop diskette operation" command is being received, output 3 from

the decoder goes true, asserting a low level on the SET STOP/ line (pin P2-31). The subsequent low-to-high transition on SET STOP/ clocks the stop latch reset (i.e., the active-low state). After the stop latch is sampled, the microprogram presets (i.e., clears to the non-active-high state) the latch.

If a "reset channel" command is being received, output 7 from the decoder goes true, asserting a low level on the RESET/ line to the Interface Board (pin P2-49).

3.2.2 MICRO CONTROL UNIT (MCU) BLOCK

The micro control unit (MCU) block provides the addresses for the microprogram memory. Since the microinstructions which are fetched from microprogram memory and executed by the central processing elements define the specific functions performed by the Channel Board, the MCU block can be considered the primary source of control for the diskette controller. In addition to an Intel 3001 Microprogram Control Unit device, the MCU block includes an 8234 eight-to-four multiplexer, a 74151 eight-to-one multiplexer and a few gating circuits, as shown on sheet 3 of the board schematic (Section 3.3).

The Intel®3001 Microprogram Control Unit controls the sequence in which microinstructions are fetched from the microprogram memory. Its functions include the following:

- Maintenance of the microprogram address register
- Selection of the next microinstruction based on the contents of the microprogram address register.
- Decoding and testing of data supplied via several input busses to determine the microinstruction execution sequence.
- Saving and testing of carry output data from the central processor (CPE) array.
- Control of carry/shift input data to the CPE array.
- Control of microprogram interrupts.

A functional block diagram of the 3001 MCU is shown in Figure 3-2.

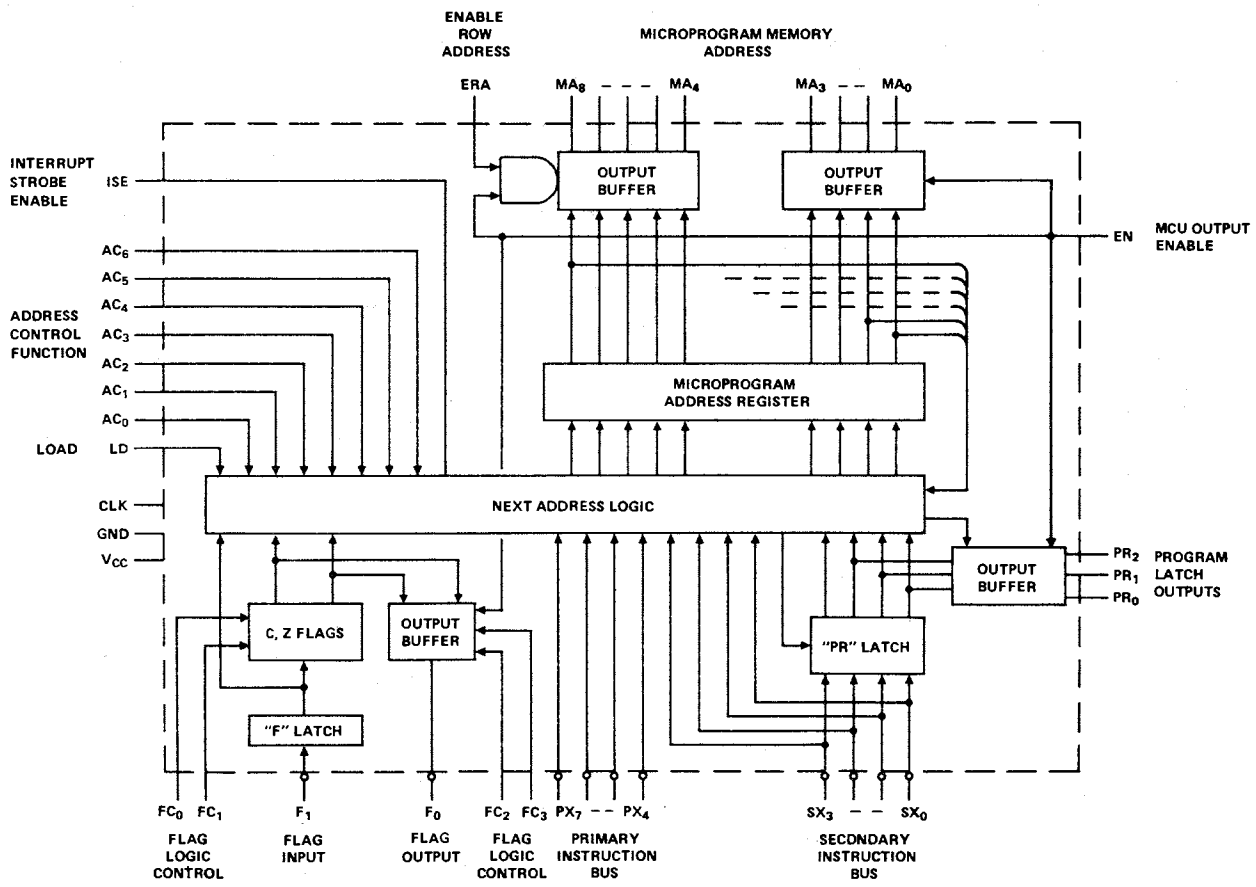


FIGURE 3-2

3001 MICROPROGRAM CONTROL UNIT: FUNCTIONAL BLOCK DIAGRAM

Address control information is supplied to the 3001 at inputs AC0-AC6. AC1-AC6 are provided directly from bits 27-32 of the microinstruction currently being fetched from microprogram memory (see Section 3.2.3). The AC0 input, however, is supplied by the output of the 74151 eight-to-one multiplexer.

Five control lines from the Interface Board (AZ, INDEX, XFER REQ, TIME OUT and F), one line from the CPE block (CO) and one line from the channel command block (BUSY START) are applied to the 74151 inputs along with the AC0 bit (bit 26) from the microinstruction currently being fetched. The three select lines applied to the A, B and C inputs on the 74151 section specify which of the eight lines are actually multiplexed through to the AC0 input on the 3001 device. The three select lines are supplied by bits 14, 15 and 16 (IN0, IN1 and IN2) of the current microinstruction. Table 3-1 correlates the values in bit positions 14, 15 and 16 of the current microinstruction with the control line which is multiplexed into the AC0 input on the 3001 section.

The "flag logic" input (F1) to the 3001 device (pin 17) is also provided by CO. The level on the CO line will reflect the "Carry out" output from the 3002 Central Processing Element (CPE) at A21 or the "shift right" output from the 3002 CPE at A23.

The "clock" input (CLK) to the 3001 device (pin 19) is supplied by CLK1/ (pin P2-3) which is one of the two clock pulses generated on the Interface Board (see Chapter 4).

The load input (LD) to the 3001 (pin 36) is fed by the master reset (MR) signal from the Interface Board (pin P2-56).

The four "flag logic control" outputs (FC0 - FC3) to the 3001 device are provided by bits 17 and 18 of the current microinstruction. Bit 17 is applied to both FC0 (pin 15) and FC1 (pin 16) while bit 18 is applied to FC2 (pin 13) and FC3 (pin 12).

TABLE 3-1

AC0 INPUT SELECTION

Select Lines			AC0 Input At 3001 MCU (Pin 39)
(Bit 14)	(Bit 15)	(Bit 16)	
0	0	0	Bit 26 of current microinstruction (AC0)
0	0	1	CO: The carry out output from the 3002 CPE at A21 or the shift right output from the 3002 $\overline{CPE}$ at A23.
0	1	0	BUSY START: Indicates that a "write MA lower", "write MA upper" or "read result byte" channel command has been received.
0	1	1	F (pin P2-58): Indicates that data/clock shift registers are full during read or empty during write.
1	0	0	AZ (pin P2-41): Indicates a valid CRC check (all zeros).
1	0	1	INDEX (pin P2-42): Indicates that an index mark has been detected (i.e., the beginning of a track).
1	1	0	XFER REQ (pin P2-39): Indicates that Interface Board has requested use of the SBC system bus.
1	1	1	TIMEOUT (pin P2-38): Ten millisecond pulse that is used by the microprogram for general timeouts.

The "enable" (EN) and "enable row address" (ERA) inputs to the 3001 (pins 25 and 35, respectively) are held high.

Unless the active-low master reset (MR/) signal from the Interface Board is true (low), the "secondary instruction bus" inputs (SX0 - SX3) to the 3001 device (pins 10, 8, 6 and 5, respectively) will reflect the complement of the level on the four least significant memory address outputs (MA0 - MA3). That is, SX0 = $\overline{MA0}$, SX1 = $\overline{MA1}$, SX2 = $\overline{MA2}$ and SX3 = $\overline{MA3}$ if MR/ is false (high). If MR/ is true (low), the four SXn inputs will all be high, regardless of the state of the MAn outputs.

The four "primary instruction bus" inputs (PX4 - PX7) to the 3001 device are fed by the four inverting outputs of the 8234 eight-to-four multiplexer (A5). The multiplexer outputs are controlled by the levels on the S0 and S1 inputs.

The A1, A2 and A3 inputs to the 8234 section are the three least significant data outputs from the central processing element (CPE) block that have been buffered and inverted; the A0 input is always high. After having fetched the diskette instruction byte from the I/O Parameter Block in system memory (see Section 2.3), the CPE block will output the three bits that specify one of the seven diskette operations onto its three least significant data outputs (D0 - D2). At this time, the mask and output bits of the current microinstruction being output from the microprogram memory block (see Section 3.2.3) will produce a high level on the S0 input to the 8234 section and a low level on the S1 input, multiplexing the inverted levels of the 8234's A inputs (specifying a particular diskette operation) into the PX4-PX7 inputs on the 3001 MCU. This allows the 3001 MCU to subsequently access those microinstructions which will effect the appropriate diskette operation.

The B1, B2 and B3 inputs to the 8234 multiplexer are the three least significant system address bits that were buffered and inverted in the channel command block (see Section 3.2.1); the B0 input is always held low. Recall that these three address bits specify one of the seven channel commands. At this time, the mask and output bits of the current microinstruction being output from the microprogram

memory block (see Section 3.2.3) will produce a low level on the S0 input to the 8234 section, multiplexing the inverted levels of the 8234's B inputs (specifying a particular channel command) into the PX4 - PX7 inputs on the 3001 MCU. This allows the 3001 MCU to subsequently access those microinstructions which will produce the proper Diskette System response to the channel command received.

The MCU outputs the 9-bit address of the next microinstruction to be fetched on MA0 - MA8. MA0 - MA8 are applied to the nine address inputs on each of the 3604 PROM's that constitute the microprogram memory (see Section 3.2.3).

3.2.3 MICROPROGRAM MEMORY BLOCK

The micro program memory block stores the microinstructions which direct the operation of the diskette controller. The microprogram memory block consists of four 3604 programmable-read-only-memory devices (512 x 8 bits each), which store 32 bit microinstructions; a 3205 three-to-eight decoder, which generates eight timing control pulses (DEC OUT0 - DEC OUT7) based on bits 11, 12 and 13 (OUT0 - OUT2) of the current microinstruction; and a 3404 six-bit high speed latch, which provides various control signal levels based on the decoder outputs mentioned above and the mask bit field of the current microinstruction; as shown on sheet 3 of the board schematic (Section 3.3).

The 9-bit memory address (MA0 - MA8) for the four 3604 PROM's is provided by the 3001 Microprogram Control Unit. MA0 - MA8 cause the addressed microinstruction to appear on the 32 output lines from the four PROM's.

Table 3-2 summarizes bit definitions for the 32-bit microinstructions. The address control bits, AC0 - AC6 (bits 26-32), the flag control bits (bits 17 and 18), and the input control bits used for AC0 select (bits 14-16) are fed to the micro control unit block as described in Section 3.2.2. The function field bits, F0 - F6

TABLE 3-2

MICROINSTRUCTION BIT ASSIGNMENTS

MICROINSTRUCTION BIT	SIGNAL	DEFINITION	PROM LOCATION - PIN
01	MASK0 (M0)		A13- 9
02	MASK1 (M1)		A13-10
03	MASK2 (M2)		A13-11
04	MASK3 (M3)	Mask field	A13-13
05	MASK4 (M4)		A13-14
06	MASK5 (M5)		A13-15
07	MASK6 (M6)		A13-16
08	MASK7 (M7)		A13-17
09	SLK0 (S0)	K-bus select	A12- 9
10	SLK1 (S1)		A12-10
11	OUT0		A12-11
12	OUT1	Decoder output select	A12-13
13	OUT2		A12-14
14	IN0		A12-15
15	IN1	AC0 select	A12-16
16	IN2		A12-17
*17	FC0		A11- 9
*17	FC1	Flag control	A11- 9
*18	FC2		A11-10
*18	FC3		A11-10
19	F0		A11-11
20	F1		A11-13
21	F2		A11-14
22	F3	Function field	A11-15
23	F4		A11-16
24	F5		A11-17
25	F6		A10- 9
26	AC0		A10-10
27	AC1		A10-11
28	AC2		A10-13
29	AC3	Address control	A10-14
30	AC4		A10-15
31	AC5		A10-16
32	AC6		A10-17

* Bit 17 is applied to both the FC0 and FC1 flag control inputs on the 3001 MCU, while bit 18 is applied to both the FC2 and FC3 inputs.

(bits 19-25), the mask bits M0 - M7 (bits 1-8) and the K-bus select bits S0 and S1 (bits 9-10) are applied to the central processing element (CPE) block, as described in Section 3.2.4. The output bits OUT0 - OUT2 (bits 11-13) are applied to the three address inputs on a 3205 decoder. The enable inputs to this 3205 section are provided by the CLK2/ pulse from the Interface Board (pin P2-30), the seventh mask bit M7 (bit 8 of the current microinstruction) and S0, one of the two K-bus select bits mentioned above (bit 9 of the current microinstruction). The eight decoder outputs, DEC OUT0 - DEC OUT7, provide timing control pulses that, when used with the mask bits, provide overall control for the diskette controller (see Table 3-3).

The mask field bits M0 - M7 (bits 1-8 of the current micro-instruction) are actually used for three disjoint functions in the Diskette System:

1) Generating control signals for the hardware. These control signals are pulses (positive or negative and of 50-75 nsec. duration) or levels. The pulses are generated by gating the appropriate mask bit with one of the 3205 decoder outputs (DEC OUTn). The levels are derived by using the DEC OUTn outputs to strobe 3404 six-bit latches. The DEC OUTn outputs provide the write enable strobes to the 3404 latches, while the mask bits provide the data inputs and, consequently, the actual controls. Refer to Table 3-3 for a summary of the control pulses and levels generated by the DEC OUTn strobes and mask field bits.

2) Driving the input multiplexers to the 3002 Central Processing Element (CPE) array. Four fields (data shift register, clock shift register, errors and status) are multiplexed through 8234 and 8233 multiplexer devices (see sheet 4 of the schematic) and into the I-bus inputs of the CPE array. The selection bits for the multiplexers are provided by the mask field bits, as listed in Table 3-4.

TABLE 3-3 CONTROL PULSES AND LEVELS GENERATED BY MICROPROGRAM

OUT CODE → (DEC OUT n) / MASK BIT ↓	000 (0)	001 (1)	010 (2)	011 (3)	100 (4)	101 (5)	110 (6)	111 (7)	FIRMWARE MNEMONICS ASSIGNED TO EACH CONTROL FUNCTION WITHIN THE MICROPROGRAM
M0	CSTEP		NSUB0				SMREQ	CSYNC	
M1	SSCLK	RDYRS	NSUB1			WFLRS	STBDU	RSAMD	
M2	LDHD			LDNXM			STBDL	UNLHD	
M3		RINH				LOWEN	LDADD	RSTST	
M4		SINH		WTGTN		MEMWT	GTR41	NGT41	
M5		SACK		SR OUT	CRC MD		SINTR	RNDX	
M6		RDOR		LDADM	DIREC		CINBS	LDOPC	

-- CONTROL FUNCTION DEFINITIONS --

CSTEP - This pulse triggers the one-shot which drives the STEP/ line to the diskette drives. STEP/ causes the selected drive to move its head one track.

SSCLK - This pulse triggers the TIMEOUT one-shot (A55). TIMEOUT provides a 10 msec. pulse for use by the microprogram.

LDHD - This pulse sets the LOAD latch on the Interface Board which causes the read/write head on the selected unit to be loaded.

RINH - This pulse resets the inhibit memory write latch (A48-1).

SINH - This pulse sets the inhibit memory write latch (A48-4).

SACK - This pulse sets the transfer acknowledge (XACK/) latch on the Interface Board.

RDOR - This pulse resets the data overrun latch (A33-13) on the Interface Board.

TABLE 3-3 (CONTINUED)

-- CONTROL FUNCTION DEFINITIONS --

NSUB0 and NSUB1 - These levels select "primary instruction bus" inputs to 3001 MCU.

LDNXM - This level is used to load clock shift register with the bit patterns required to write the different address marks onto a diskette.

WTGTN - When the data and clock shift registers are empty, this level allows both shift registers to be parallel loaded with information that will be serially shifted out to the Interface Board and then to the diskette. Used to generate PE/ and WRT GT/ signals.

SROUT - This level enables data bits from data register (when high) or from 8503 CRC device (when low) to be sent to the selected diskette.

LDADM - This level is used to load clock shift register with the bit patterns required to write the different address marks onto a diskette.

CRCMD - This level indicates the operating mode for the 8503 CRC device (A11) on the Interface Board.

DIREC - This level indicates the direction of head movement for the selected diskette drive.

LOWEN - This level drives the GATE LOWER line that enables data onto the memory data inputs (MD0/-MD7/) to the CPE array.

MEMWT - This level indicates when the Diskette System wishes to write data to memory.

SMREQ - This pulse initiates the bus request sequence intended to gain master control of the SBC system bus.

STBDU - This pulse loads CPE data outputs (D0-D7) into the latch which drives the system data lines, DATA8/-DATAF/.

STBDL - This pulse loads CPE data outputs (D0-D7) into the latch which drives the system data bus lines, DATA0/-DATA7/.

LDADD - This pulse loads CPE data outputs (D0-D7) into the latch which drives the system address bus lines, ADR0/-ADR7/.

TABLE 3-3 (CONTINUED)

-- CONTROL FUNCTION DEFINITIONS --
SINTR - This pulse is used to generate a signal which clocks the interrupt latch on the Channel Board.
CINBS - This pulse is used to latch data from the system data bus into the latches which drive MØ-M7/ of the CPE array.
CSYNC - This pulse initializes the synchronization logic prior to detecting an address mark.
RSAMD - This pulse resets the synchronization logic prior to initializing the logic with the CSYNC pulse.
UNLHD - This pulse clears the LOAD latch on the Interface Board, and ultimately causes the read/write head on the selected drive to be unloaded.
RSTST - This pulse is used to generate CLR START STOP signal which resets the STOP latch and the 74145 latch at A8-1 on the Channel Board.
RNDX - This pulse resets the INDEX latch (A27-13) on the Interface Board.
LDOPC - This pulse latches diskette operation code (D0-D5) and makes latched code available to "primary instruction bus" multiplexer.
RDYRS - This pulse resets the Drive Ready flip-flops on the Interface Board.
WFLRS - This level generates the write fault reset pulse to the selected drive.
GTR41 - This pulse sets the low current mode for writes on tracks greater than 41_{10}.
NGT41 - This pulse resets the low current mode for writes on tracks equal to or less than 41_{10}.

TABLE 3-4

I-BUS SELECTION BY MASK FIELD BITS

	INPUT FIELDS*			
MASK BITS	DATA SHIFT REGISTER (A36 and A29) Outputs	CLOCK SHIFT REGISTER (A34 and A27) Outputs	ERROR LINES (DOR, WRT PROT, WRT ERR, SEL DR NRDY)	STATUS LINES (DR0, DR1, STOP, TRACK00)
M7 =	1	1	1	1
M6 =	0	0	1	1
M5 =	0	0	0	1
M4 =	1	1	0	0
M3 =	0	1	0	0

* An input field will be multiplexed into the I inputs of 3002 CPE array if the mask bits reflect the values listed for that field and if the K-Bus select line, S0 (bit 9 of the current microinstruction) is high (logical 1). Refer to Section 3.2.4 for a more complete description of the 3002 CPE array inputs.

3) Providing generalized inputs to the K-bus inputs of the 3002 CPE array. The mask bits are multiplexed through two 8234 multiplexer devices and into K-bus inputs of the CPE array. The selection bits are provided by S0 and S1 (bits 9-10 of the current microinstruction). Table 3-5 correlates the levels on the S0 and S1 lines with the K-bus inputs.

3.2.4 CENTRAL PROCESSING ELEMENT (CPE) BLOCK

The central processing element (CPE) block executes the function indicated by each microinstruction output from the microprogram memory. The CPE block includes an array of four Intel®3002 Central Processing Elements, as well as four 8234 and one 8233 eight-to-four multiplexers that provide various inputs to the CPE array, as shown on sheet 4 of the board schematic (Section 3.3).

An Intel 3002 Central Processing Element contains all of the circuits that represent a 2-bit wide slice through the data processing section of a digital computer. When wired together in an array, a set of CPE's provide the following capabilities:

- Two's complement arithmetic
- Logical AND, OR, NOT and exclusive-OR
- Incrementing and decrementing
- Shifting left or right
- Bit testing and zero detection
- Carry look-ahead generation
- Multiple data and address busses

A functional block diagram of a 3002 CPE is shown in Figure 3-3.

TABLE 3-5

K-BUS INPUT SELECTION

K-BUS SELECT LINES		K-BUS INPUTS TO 3002 CPE ARRAY							
S1 (Bit 10)	S0 (Bit 9)	K0	K1	K2	K3	K4	K5	K6	K7
0	0	M0	M1	M2	M3	M4	M5	M6	M7
0	1	1	1	1	1	1	1	1	1
1	1	0	0	0	0	0	0	0	0

Note that only when S0 and S1 equal zero will the K-bus inputs be supplied by the mask field bits; otherwise, the K-bus inputs will be all ones or all zeros.

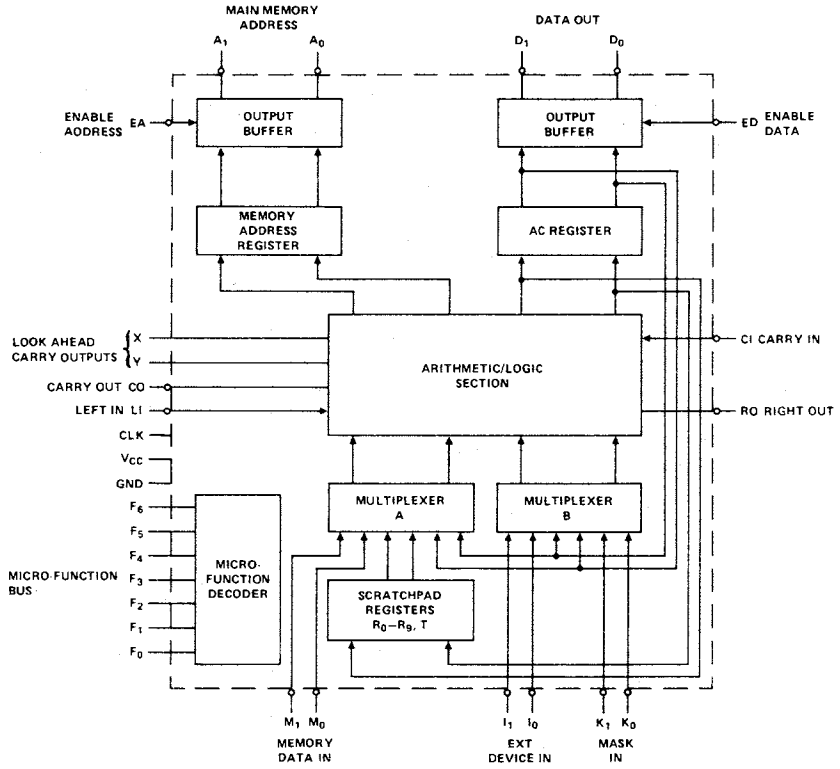


FIGURE 3-3

3002 CENTRAL PROCESSING ELEMENT: FUNCTIONAL BLOCK DIAGRAM

During each micro-cycle, the function field of the current microinstruction is applied to the F-bus inputs (F0-F6) of each 3002 CPE. The function bits are decoded, the operands are selected by the internal multiplexers and the specified operation is performed. Within each CPE, data is stored in eleven scratchpad registers or the accumulator.

Data being output from the CPE array is carried on the address bus (A0-A7) or the data out bus (D0-D7). The address outputs drive the eight most significant lines of the SBC system address bus, ADR8 - ADRF. The data outputs are made available to the data flow control block (Section 3.2.6) and the micro control unit block (Section 3.2.2).

Data is brought into the CPE array on three separate input buses, the memory data bus (M0-M7), the I-bus (I0-17) and the K-bus (K0-K7). The memory data inputs are supplied by the data flow control block (Section 3.2.6) which routes data from the system data bus to the CPE array. The latter two buses are driven by five multiplexers.

The K-bus inputs (K0-K7) are driven by two 8234 eight-to-four multiplexers (located at A33 and A26). The S0 and S1 outputs from the microprogram memory (bits 9 and 10 of the current microinstruction) determine the four outputs on each of the two 3234 sections. If S0=1 and S1=0, all eight inverted ouptuts will be low. If S0=1 and S1=1, all eight outputs will be high. If S0=0, however, the eight mask bits, M0-M7, from the microprogram memory (bits 01-08 of the current microinstruction) will be inverted and applied to the K-bus inputs of the CPE array (see Table 3-5).

The I-bus inputs (I0-I7) are driven by two 8234 (inverting outputs) and one 8233 (non-inverting outputs) eight-to-four multiplexers (located at A28, A45 and A35). Mask bits M3-M7 from the microprogram memory (bits 04-08 of the current microinstruction) determine the I-bus inputs as listed in Table 3-4 of Section 3.2.3. These mask bits will enable the eight outputs from the two 4-bit data shift

registers (at A36 and A29), or the eight outputs from the two clock shift registers (at A34 and A27) through the 8233 section and one of the 8234 sections and into the I0-I7 inputs on the CPE array. The mask bits can also enable the four error lines (DOR, WRT PROT, WRT ERR and SEL DR NRDY) or the four status lines (DR0, DR1, STOP and TRACK 00) from the Interface Board (see Chapter 4) through the other 8234 section (at A45) and into the I4-I7 inputs on the CPE array.

The clock input to the 3002 CPE's is provided by the CLK1/ pulse generated on the Interface Board. The enable address inputs (EA), which enable the address outputs (A0-A7), are fed by the SELECTED line from the Interface Board (pin P2-43). When true, SELECTED indicates that the diskette controller has master control of the SBC bus (see Chapter 4). The enable data input (ED) to each 3002 CPE is permanently held low (active).

The carry output (CO) from each 3002 CPE feeds the carry input (CI) of the adjacent 3002 CPE. Likewise, the shift right output (RO) feeds the shift right input (LI) of the adjacent 3002 CPE. The carry output (CO) from the most significant CPE (A21-7) feeds its own shift right input (LI) and is wire-ORed with the shift right output (RO) from the least significant CPE (A23-8) to form the CO line, which is applied to the flag control input (FI) on the 3001 Micro-program Control Unit. CO is also inverted and applied to the D1 input on the 74151 multiplexer (A30-3) which provides the least significant address control bit, AC0, to the 3001 MCU. The carry input (CI) to the least significant 3002 CPE (A23-10) is provided by the flag control output (FO) from the 3001 MCU.

3.2.5 DATA/CLOCK SHIFT REGISTER BLOCK

During read operations, the data/clock shift register (SR) block accepts serial data and clock bits that the Interface Board has received from the selected diskette drive and converts them into eight bit bytes that are input, in parallel, to the CPE array. During write operations, the shift registers are parallel loaded with data and clock bytes which are then shifted out to the Interface Board. This block includes four 9300 four-bit shift registers, as shown on sheet 4 of the board schematic (see Section 3.3).

Recall from Section 1.2 that data and clock bits are interspersed when information is written on disk. When that information is later read, the diskette drive separates the data and clock pulses before passing them to the Interface Board (see Chapter 4).

The Interface Board, in turn, sends the data bits to the Channel Board via the SR DATA IN/ line (pin P2-22) along with the data strobe SR DATA STB/ (pin P2-24). SR DATA IN/ feeds the J and $\overline{K}$ inputs on the first data shift register at A36. The Q3 output from this first shift register then feeds the J and $\overline{K}$ inputs on the shift register at A29. When the data shift registers are full, the eight data bits are transferred in parallel to the CPE array. The four outputs from the first data shift register (A36) are applied to the "A" inputs on the 8233 multiplexer that feeds the four least significant I-bus inputs to the CPE array. The first two outputs are also made available to the Interface Board via the ID0/ (pin P2-26) and ID1/ (pin P2-23) lines (see Chapter 4). The four outputs from the second data shift register (A29) are applied to the 8234 multiplexer (A28) that feeds the four most significant bits of the I-bus inputs to the CPE array. Both data shift registers are clocked by SR DATA STB.

During write operations, both data shift registers can be parallel loaded when the PE/ input is low. PE/ will go low when the shift register is empty and the WTGTN signal is true (see Table 3-3).

The parallel inputs to the first data shift register (A36) are supplied by the four least significant data outputs from the CPE array, as follows:

D0 ⟶ P0
D1 ⟶ P1
D2 ⟶ P2
D3 ⟶ P3

The parallel inputs to the second data shift register (A29) are supplied by the four most significant CPE array data outputs:

D4 ⟶ P0
D5 ⟶ P1
D6 ⟶ P2
D7 ⟶ P3

The data bits can then be shifted out, in serial, to the Interface Board via the SR DATA OUT line (pin P2-25) which is driven by the $\overline{Q3}$ output on the second data shift register.

During read operations, the Interface Board sends the clock bits (that were interleaved with the data bits on the diskette) to the Channel Board via the SR CLK IN/ line (pin P2-6) along with the clock strobe, CLK SR STB (pin P2-8). SR CLK IN/ feeds the J and $\overline{K}$ inputs on the first clock shift register at A34. The Q3 output of this first clock shift register then feeds the J and $\overline{K}$ inputs on the shift register at A27. When the clock shift registers are full, the eight clock bits can be transferred in parallel to the CPE array. The four outputs from the first clock shift register (A34) are applied to the "B" inputs on the 8233 multiplexer that feeds the four least significant bits of the I-bus to the CPE array. The four outputs from the second clock shift register (A27) are applied to the 8234 multiplexer that feeds the four most significant bits of the I-bus. Both clock shift registers are clocked by CLK SR STB.

During write operations, both clock shift registers can be parallel loaded when the PE/ signal (described above) is low. The P0, P1 and P2 inputs to the first clock shift register and the P2 and P3 inputs to the second clock shift register are tied to ground. The P3 input to the first shift register and the P1 input to the second are both fed by the output of a 7408 AND gate (A7-6) shown on sheet 3 of the schematic. The inputs to this gate are the LDNXM and LDADM control levels which have been set in the 3404 latches at A16 by the microprogram (refer to Table 3-3). The P0 input to the second clock shift register is fed by the LDADM control output. LDNXM and LDADM allow the microprogram to produce the varied patterns of clock bits that are required to write the different types of address marks onto a diskette (refer to Section 1.2).

3.2.6 DATA FLOW CONTROL BLOCK

The data flow control block routes data to/from the various other functional blocks within the Channel Board. This block consists of five 8212 bi-directional latching bus drivers, a 3404 six-bit latch and various gating circuits, as shown on sheet 2 of the board schemati (Section 3.3).

The six least significant data outputs (D0-D5) from the CPE array are applied to the inputs of the 3404 six-bit latch. When the micro-program generates the LDOPC control pulse (see Table 3-3), D0-D5 are latched and inverted. $\overline{D0}$-$\overline{D2}$ are made available to the "primary instruction bus" multiplexer (A5) in the MCU block. $\overline{D3}$ is used for 16-bit data flow control as described below. $\overline{D4}$ and $\overline{D5}$ are sent to the Interface Board as the unit select signals, USA (pin P2-9) and USB (pin P2-12), respectively.

All eight CPE data outputs, D0-D7, are also applied to three of the 8212 latching bus drivers. DØ-D7 are loaded into the 8212 device at A41 when the microprogram generates the LDADD control pulse (see Table 3.3). The 8212 device at A41 drives the eight least significant lines

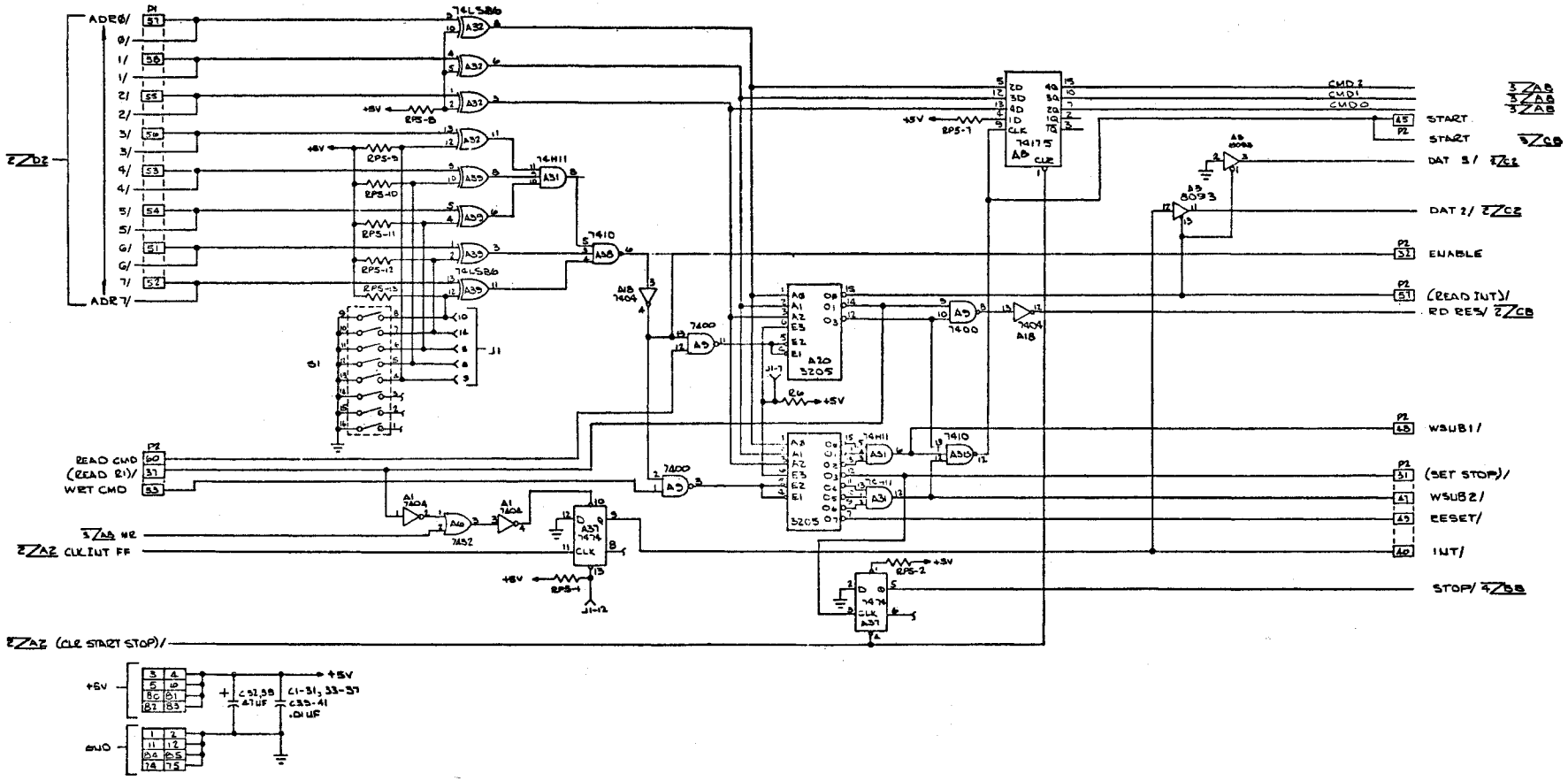
of the system address bus, ADRØ/-ADR7/. When the microprogram generates the STBDL control pulse (see Table 3-3), DØ-D7 are latched into the 8212 device at A43 which drives the eight least significant lines of the system data bus, DATØ1-DAT7/ (pins P1-67 through P1-74). STBDL is generated when the microprogram is setting a result word or preparing write data. When the microprogram generates the STBDU control pulse (see Table 3-3), DØ-D7 are latched into the 8212 device at 42, which drives the eight most significant system data bus lines, DAT8/-DATF/ (pins P1-59 through P1-66). STBDU would only be generated if the Diskette System were operating in the 16 bit mode. The address latch (A41) is gated onto the SBC system bus whenever the SELECTED signal is true. The high-order data latch (A42) is gated onto the bus whenever the SELECTED, MEMORY WRITE and 16-BIT MODE signals are true. The low-order data latch (A43) is gated onto the bus during memory write operations or "read result" operations.

When a master bus module in the SBC outputs data to the diskette controller, the data from lines DATØ/-DAT7/ is applied to the 8212 bus driver at A25. The data on those lines are strobed into the input latches at A24 and A25 by the microprogram control pulse CINBS (see Table 3-3). "Slave" memory modules also drive the data lines, DATØ/-DAT7/ and possibly DAT8/-DATF/. Memory data are strobed into the latches when the STB MEM IN signal is generated by the Interface Board. The outputs from the devices at A24 and A25 are "OR-ed" into lines MDØ/-MD7/. Depending on the state of the GATE LOWER signal, data from only one of the two latches will actually be gated onto the M input lens (MDØ/-MD7/) to the CPE array.

3.3 SCHEMATICS/PIN LISTS: CHANNEL BOARD

The schematic drawing (4 sheets) for the Channel Board is provided in Figure 3-4. These schematics reflect the latest revision level of the board. Appendix A contains a copy of the assembly drawing for the board; the assembly drawing should be consulted for the location of the various devices on the board.

Table 3-6 lists the pins and designated signal functions for the 86-pin P1 bus connector. Table 3-7 lists the same information for the 60-pin P2 controller connector.



NOTES: UNLESS OTHERWISE SPECIFIED.

1. THIS DOCUMENT REFLECTS ARTWORK NO. 1000465, REV 'C'.
2. RESISTOR VALUES ARE IN OHMS 1K, 1/4W, 5%.
3. CONNECTOR J1 IS USED FOR TEST POINT ONLY.

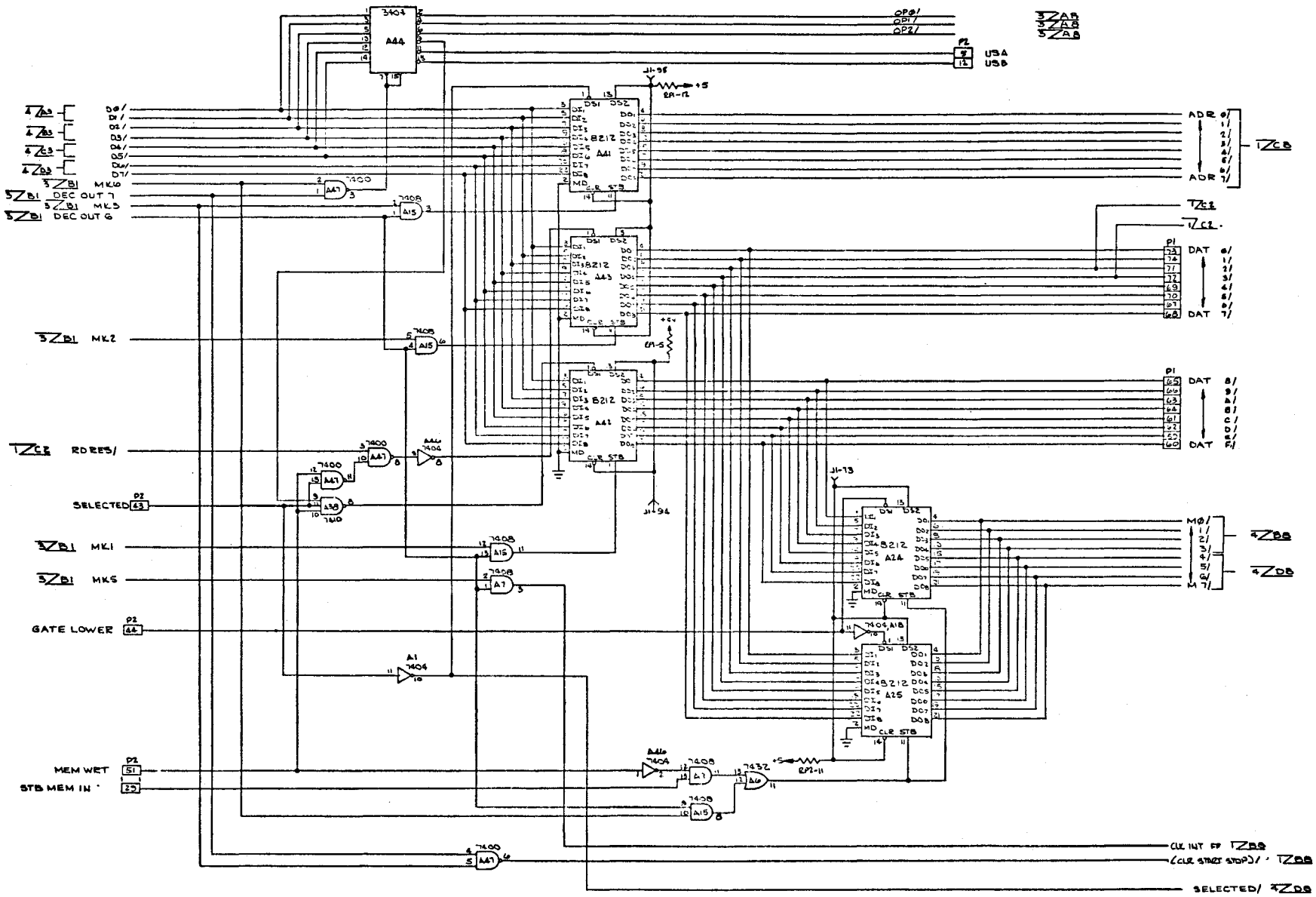
DESCRIPTION

PARTS LIST

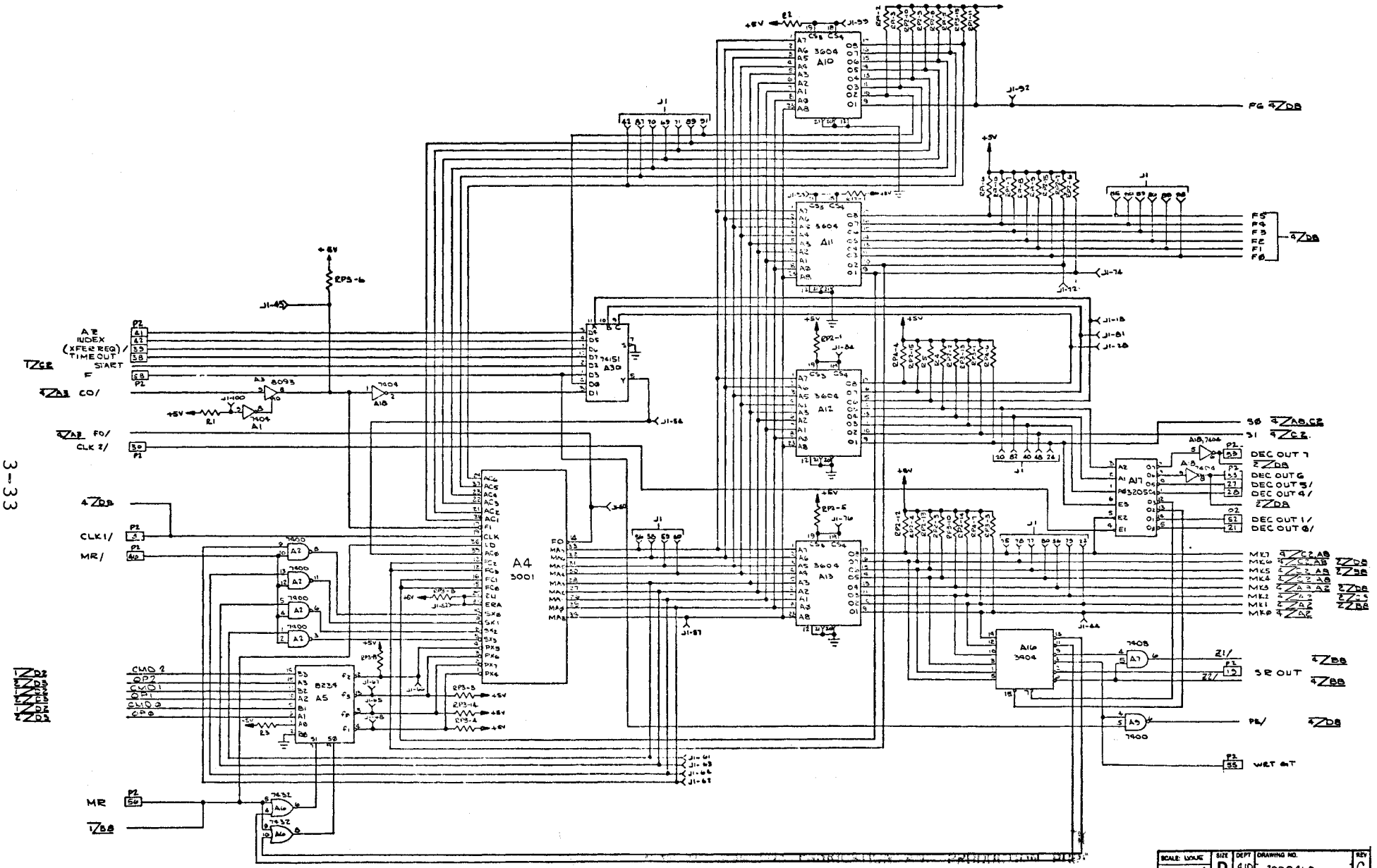
intel © INTEL 1977 3065 BOWERS AVE. SANTA CLARA CALIF. 95051

TITLE: SCHEMATIC FLOPPY DISK CONTROLLER CHANNEL

SIZE	DEPT	DRAWING NO.	REV
D	410	1000465	C



SCALE: LOGIC	SIZE	DEPT	DRAWING NO.	REV
SHEET 1 OF 1	D	410	2000469	C



SCALE: NONE	SIZE D	DEPT 410	DRAWING NO. 2000469	REV C
SHEET 3 OF 4				

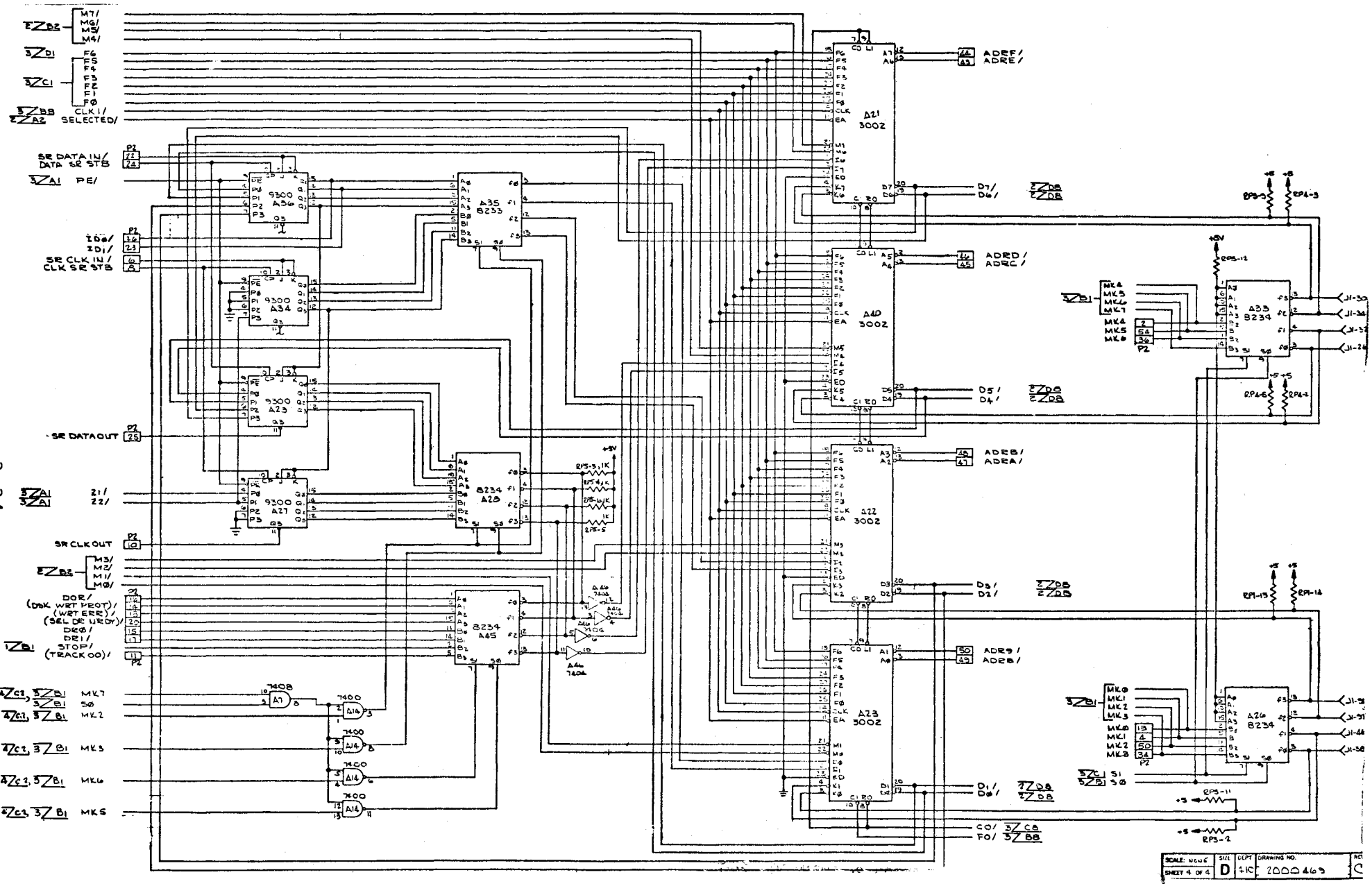


TABLE 3-6
PIN LIST: P1 BUS CONNECTOR

PIN	SIGNAL	FUNCTION
1		
↓		
42		
43	ADRE/	SBC system address bus
44	ADRF/	
45	ADRC/	
46	ADRD/	
47	ADRA/	
48	ADRB/	
49	ADR8/	
50	ADR9/	
51	ADR6/	
52	ADR7/	
53	ADR4/	
54	ADR5/	
55	ADR2/	
56	ADR3/	
57	ADR0/	
58	ADR1/	
59	DATAE/	SBC system data bus
60	DATAF/	
61	DATAC/	
62	DATAD/	
63	DATAA/	
64	DATAB/	
65	DATA8/	
66	DATA9/	
67	DATA6/	
68	DATA7/	
69	DATA4/	
70	DATA5/	
71	DATA2/	
72	DATA3/	
73	DATA0/	
74	DATA1/	
75		
↓		
86		

TABLE 3-7

PIN LIST: P2 CONTROLLER CONNECTOR

PIN	SIGNAL	FUNCTION
1	MK4	Mask bit 4
2		
3	CLK1/	Diskette controller clock 1
4	MK1	Mask bit 1
5		
6	SR CLK IN 1	Serial clock input line
7		
8	CLK SR STB	Serial clock strobe
9	USA	Unit select bit
10	SR CLK OUT	Serial clock out line
11	TRACK 00/	Track 00 detected
12	USB	(Not used)
13	MKØ	Mask bit 0
14	DSK WRT PROT/	Disk write protected
15	DRO/	Drive 0 ready
16	DOR/	Data overrun error
17	DR1/	Drive 1 ready
18	WRT ERR/	Write error
19	SR OUT	Write data multiplexer control level
20	SEL DR NRDY/	Selected drive not ready
21	DEC OUT 0/	Control decoder output 0
22	SR DATA IN/	Serial data in line
23	ID1/	Input data bit 1
24	DATA SR STB/	Serial data strobe
25	SR DATA OUT	Serial data out line
26	ID0/	Input data bit 0
27	DEC OUT 5/	Control decoder 5 output
28	DEC OUT 4/	Control decoder 4 output
29	STB MEM IN	Strobe memory data in
30	CLK2/	Diskette controller clock 2

Table 3-7. (Continued)

PIN	SIGNAL	FUNCTION
31	SET STOP/	"stop diskette" channel command
32	ENABLE	Diskette controller addressed
33	DEC OUT 7	Control decoder output 7
34	MK3	Mask bit 3
35	DEC OUT 6	Control decoder output 6
36	MK6	Mask bit 6
37	RD RI/	"read result type" channel command
38	TIME OUT	10 msec. timing pulse
39	XFER REQ/	Controller requests SBC bus
40	INT/	Interrupt line
41	AZ	All zeros, valid CRC check
42	INDEX	Index mark detected
43	SELECTED	Controller has control of SBC bus
44	GATE LOWER	Input low-order data byte
45	BUSY START	Microprogram responding to channel command
46	MR/	Master reset
47	WSUB2/	Not used at present
48	WSUB1/	Not used at present
49	RESET/	"Reset" channel command
50	MK2	Mask bit 2
51	MEM WRT	Write data to SBC memory
52	DEC OUT 1/	Control decoder output 1
53	WRT CMD	I/O write command
54	MK5	Mask bit 5
55	WRT GT	Write gate control level
56	MR	Master reset
57	RD INT/	"Read subsystem status" channel command
58	F	Shift registers full or empty
59		
60	RD CMD	I/O read command

CHAPTER 4
THE INTERFACE BOARD

The Interface Board provides the Diskette Channel with a means of communicating with the diskette drives, as well as with the Intel®system bus. Under control of the microprogram being executed on the Channel Board, the Interface Board generates those signals which cause the read/write head on the selected drive to be loaded (i.e., to come in contact with the diskette platter), then cause the head to move to the proper track. The Interface Board accepts the data being read off the diskette, interprets certain synchronizing bit patterns, checks the validity of the data using a cyclic redundancy check (CRC) polynomial, and passes the data to the Channel Board.

During write operations, the Interface Board outputs the data and clock bits to the selected drive at the proper times. It also generates CRC characters which are appended to the data; this allows the data to be verified when it is subsequently read.

When the Diskette Channel requires access to Intel®system memory, the Interface Board requests and maintains master control of the system bus, and generates the appropriate memory command.

When a CPU in the computer system issues a channel command to the diskette controller, the Interface Board acknowledges the command as required by Intel®bus protocol.

The Interface Board resides within System 80 backplane, SBC 604/614 cardcage, or a custom backplane. The Interface Board, together with the Channel Board, constitute what we refer to as the Diskette Controller.

4.1 FUNCTIONAL ORGANIZATION OF THE INTERFACE BOARD

For descriptive purposes, the circuitry on the Interface Board can be divided into five functional blocks:

- Disk drive control block
- Serial data/clock synchronization block
- Write clock generator block
- Cyclic Redundancy Check (CRC) block
- Bus control block

as shown in Figure 4-1.

The disk drive control block provides the unit selection/head loading (SELn/) signal, the direction indicator (DIR/) and the step pulse (STEP/) that moves the read/write head on the selected unit one track in the specified direction. The disk drive control block also monitors the READY status, the INDEX indicator and the TRACK0 indicator from the two drives.

The serial data/clock synchronization block receives the separated data and clock bits, and examines the bit patterns looking for specific patterns which indicate an address mark. Address marks precede address and data fields and are used to synchronize the controller with the drive. The synchronization block then generates data and clock strobes (DATA SR STB and CLK SR STB) which shift the data and clock bits into the shift registers on the Channel Board. The synchronization block also includes a bit counter that determines when a byte (8-bits) has been shifted to/from the selected drive.

The write clock generator block provides timing references for the writing of data and clock bits. Data and clock bits are both output to the drive via the WRT DAT/ line.

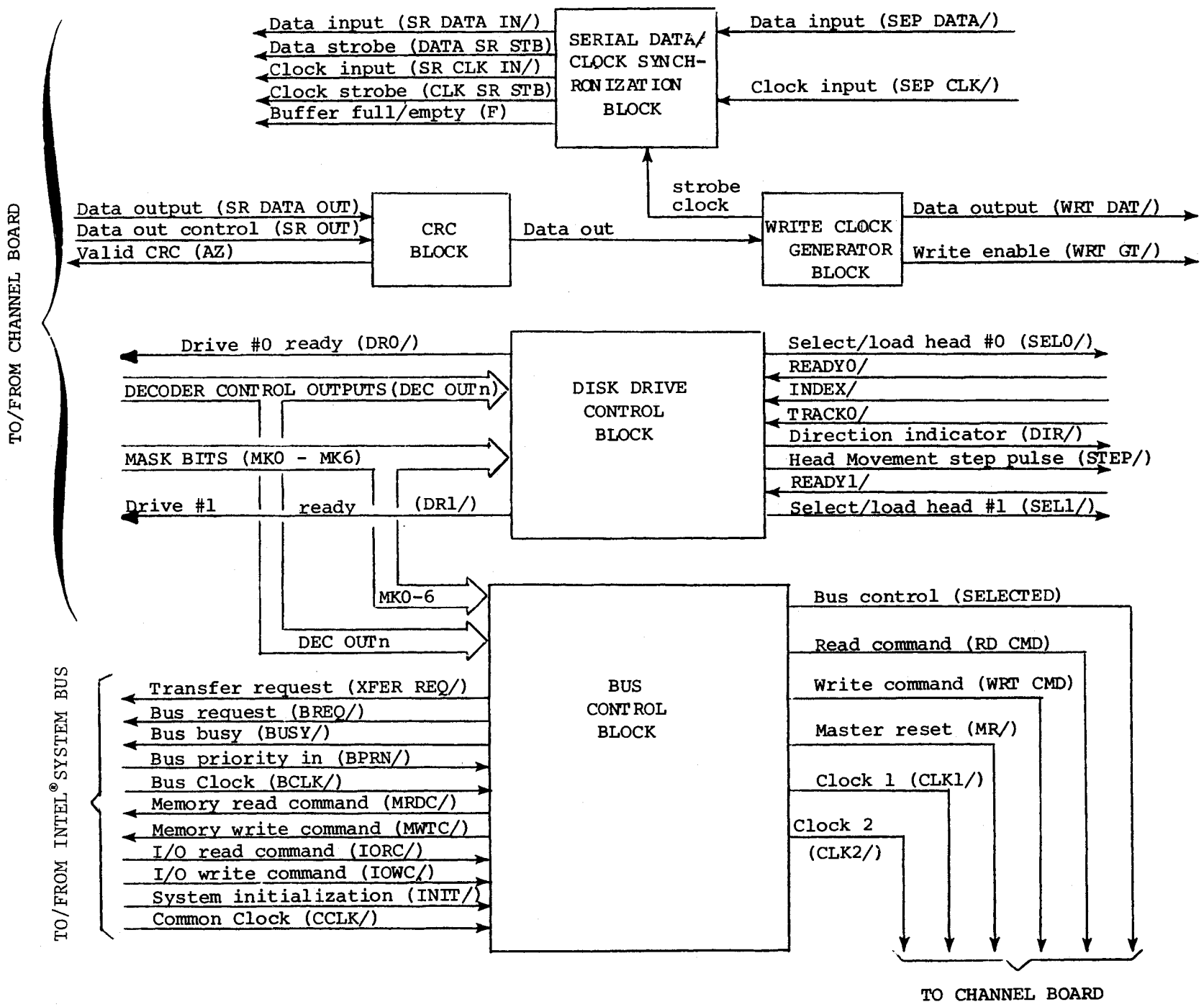


FIGURE 4-1 INTERFACE BOARD: FUNCTIONAL BLOCK DIAGRAM

The CRC block generates two CRC characters (16 bits) which are appended to the end of each address field and data field during format and write data operations. During read operations, the CRC block generates two CRC characters for each data field (includes address mark and 128 bytes of data) read, then compares these with the two CRC characters that were appended to the data field, to verify the validity of the data.

The bus control block provides the interface with the SBC system bus. The bus control block requests and maintains master control of the system bus, and generates the memory read (MRDC/) and memory write (MWTC/) commands that allow the diskette controller to access system memory. In addition, the bus control block acknowledges (XACK/) the I/O read (IORC/) or I/O write (IOWC/) command that is issued when the CPU in the SBC system executes a channel command to the diskette controller.

4.2 THEORY OF OPERATION: INTERFACE BOARD

In this section we will describe the circuitry on the Interface Board. We will divide this theory of operation discussion into five sub-sections, each dealing with one of the functional blocks defined in Section 4.1.

The Interface Board accepts/transmits signals, data and power through three different PC edge connectors:

- P1 Bus connector (to/from SBC bus)
- P2 Controller connector (to/from Channel Board)
- J1 Drive connector (to/from diskette driver)

To avoid any ambiguity when referring to connector pins in subsequent paragraphs, we will always list the connector as well as the pin whenever such references are required; for example, P1-57 refers to pin 57 on connector P1. Pin lists for the three connectors are provided in Section 4.3

The schematic drawing (4 sheets) for the Interface Board is also provided in Section 4.3.

The circuits which drive signals to the diskette drives and those which receive signals from the drives have been specified by drive manufacturer. Refer to Chapter 5 for more information concerning the electrical characteristics of the interface.

4.2.1 DISK DRIVE CONTROL

The disk drive control block generates those interface signals which cause the read/write head on the diskette drive to move to the next track (in either direction). This block consists of two 7474 D-type flip-flops, a 7451 multiplexer, two 9602 one-shot multivibrators, and assorted gating and inverting circuits, as shown on sheet 2 of the board schematic (Section 4.3).

The ready inputs from the two drives, READY1/ (pin J1-6) and READY0/ (pin J1-7) are latched and passed to the Channel Board as DR1/ (pin P2-17) and DR0/ (pin P2-15). READY1/ and READY0/ are also applied to the 7451 multiplexer. If the unit select line (USA) from the Channel Board is high, READY1 is gated through the 7451 section; if USA is low, READY0 is gated through. The inverting output of the multiplexer is inverted by a 7404 section which drives the selected drive not ready line (SEL DR NRDY/) to the Channel Board. If the ready input from the selected unit is false, then SEL DR NRDY/ will be true (low).

The unit select line (US1) which, when high, indicates that drive #1 has been selected and the complement of US1 which, when high, indicates that drive #0 has been selected are both gated together with the output of the LOAD latch (A39-5) to enable the write data (WRT DAT/, pin J1-3), the write gate (WRT GT/, pin J1-28), and the select lines, (SEL1/ and SEL0/, pins J1-30 and J1-38) to the two drives. SEL1/ or SEL0/ causes the read/write head on the selected unit to be loaded (i.e, brings the diskette into contact with the head); thus enabling circuitry in the selected drive.

The LOAD latch is set by the LDHD control pulse (A39-3) which is generated by the microprogram being executed on the Channel Board (see Table 3-3). The LOAD latch is cleared by the UNLHD pulse (A39-1) which is also generated by the microprogram.

After loading the read/write head on the selected drive, the head must be positioned over the proper track. Any of the seven diskette operations will cause the diskette controller to seek the track specified in the I/O Parameter Block (IOPB), prior to actually performing the operation (refer to Chapter 2).

The direction of head movement is defined by the level on the DIR/ line (pin J1-19). The DIREC control level, maintained by the microprogram (see Table 3-3), is applied to the 7438 NAND gate in the disk drive control block. The output from this 7438 gate (A20-11) drives the DIR/ line (pin J1-19).

Each pulse on the STEP/ line (pin J1-23) will cause the read/write head on the selected unit to move one track either in or out depending on the level on the DIR/ line. When DIR/ is high, the head will move one track away from the center of the diskette. When DIR/ is low, the head will move one track closer to the center.

The STEP/ pulse is defined by the output of a 9602 one-shot (at A21-6). This one-shot is triggered by the CSTEP control pulse, generated by the microprogram (see Table 3-3), unless the read/write head at the selected unit is already over track 0 (the outermost track) while the DIR/ line indicates outward movement. If the head on the selected unit is loaded, the output from the one-shot will produce a 100 μsec. pulse on the STEP/ line, as shown in Figure 4-2.

After the head has been positioned over the proper track (by pulsing STEP/ the required number of times), the diskette controller must wait at least 20 msec. before it begins examining the read data in an attempt to detect the ID address mark which precedes an address field. Reading the address field will verify that the seek operation placed the head over the proper track. Read initiate timing is illustrated in Figure 4-3.

The disk drive control block also includes a 9602 one-shot (at A55) which produces a 10 msec. TIMEOUT pulse, which is made available to the Channel Board (pin P2-38) for use by the microprogram. The microprogram triggers this one-shot by generating the SSCLK control pulse (see Table 3-3). Ten milliseconds after SSCLK triggers the one-shot, a low-to-high transition from the $\overline{Q}$ output of the one-shot will appear on the TIMEOUT line.

The disk drive control block accepts the TRACK0/ and INDEX/ lines from the diskette drives and passes them to the Channel Board for use by the microprogram. TRACK0/ (pin J1-18) is merely inverted twice and output as TRACK00/ at pin P2-11. INDEX/ (pin J1-2) is inverted and clocks a 7474 latch to the set state. The Q output of this latch drives the INDEX line (pin P2-42) to the

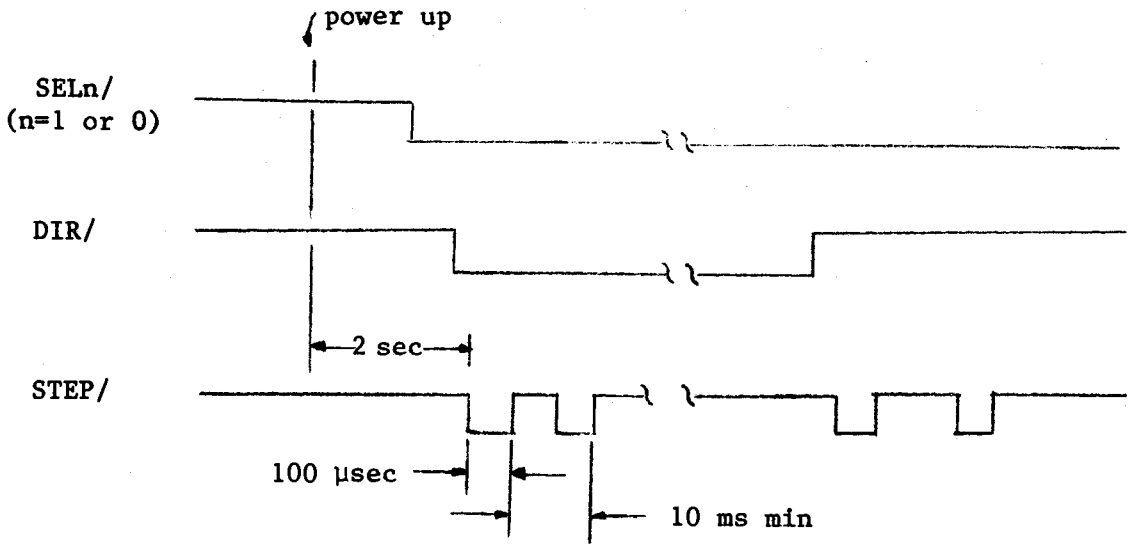


FIGURE 4-2
HEAD MOVEMENT CONTROL TIMING

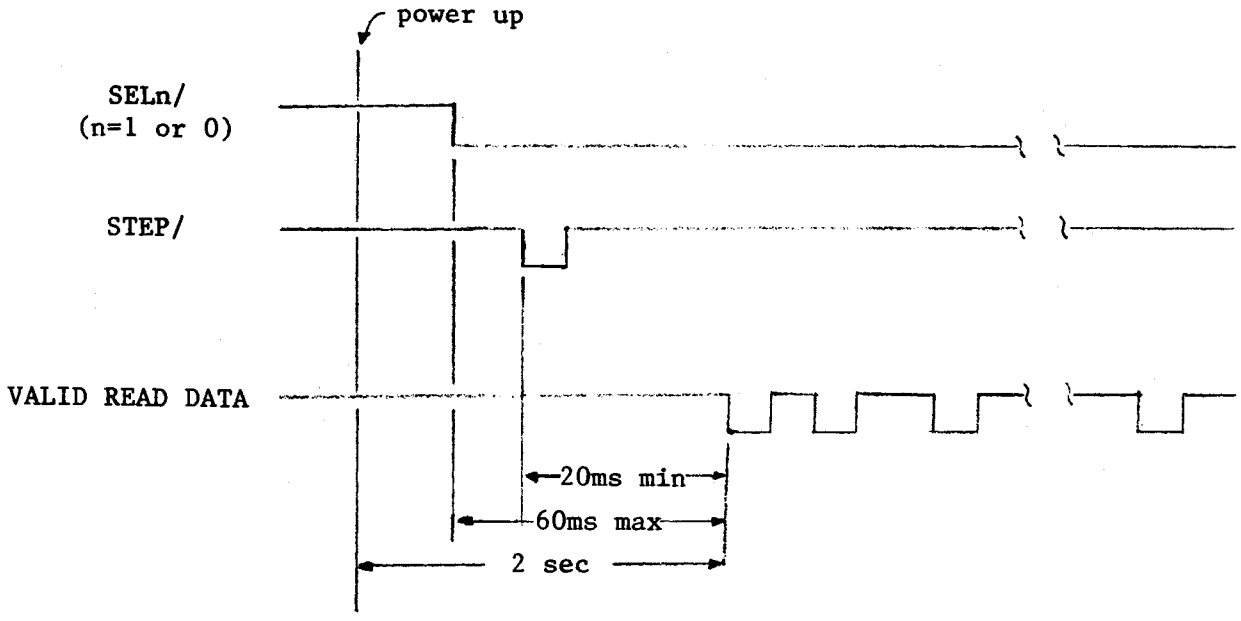
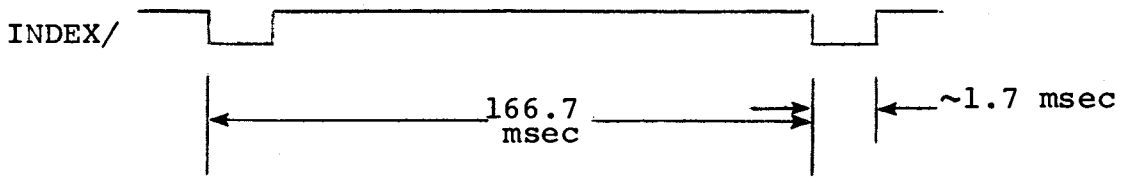


FIGURE 4-3
READ INITIATE TIMING

Channel Board. The microprogram can reset the INDEX latch by generating the RNDX control pulse (see Table 3-3). An INDEX/ pulse will be received once every 166.7 msec. and will be approximately 1.7 msec. wide:



4.2.2 SERIAL DATA/CLOCK SYNCHRONIZATION

The serial data/clock synchronization block is responsible for accepting the data and clock bits being read and detecting the beginning of address marks which were written onto a diskette when it was formatted. The address marks provide a means of synchronizing the diskette controller with the diskette platter (refer to Section 1.2). This block also includes the bit counter. The logic in this block includes two 9602 one-shot multivibrators, seven 7474 D-type flip-flops, a 74191 binary counter, a 9316 binary counter and assorted gating circuits, as shown on sheet 1 of the board schematic (see Section 4.3).

Recall from Section 1.2 that data bit cells are interleaved with clock bit cells on the diskette platter. When the diskette drive reads the platter, however, the data and clock bits are separated and sent to the Interface Board on two lines, the separated data line (SEP DATA/) and the separated clock line (SEP CLK/); refer to Figure 4-4.

An active-low data bit on SEP DATA/ is received at pin J1-15, inverted twice, and applied to the pre-set input on a 7474 latch (A31-10). An active-low clock bit on SEP CLK/ is received at pin J1-10 and inverted. The active-high SEP CLK signal triggers a 9602 one-shot (A4-4),

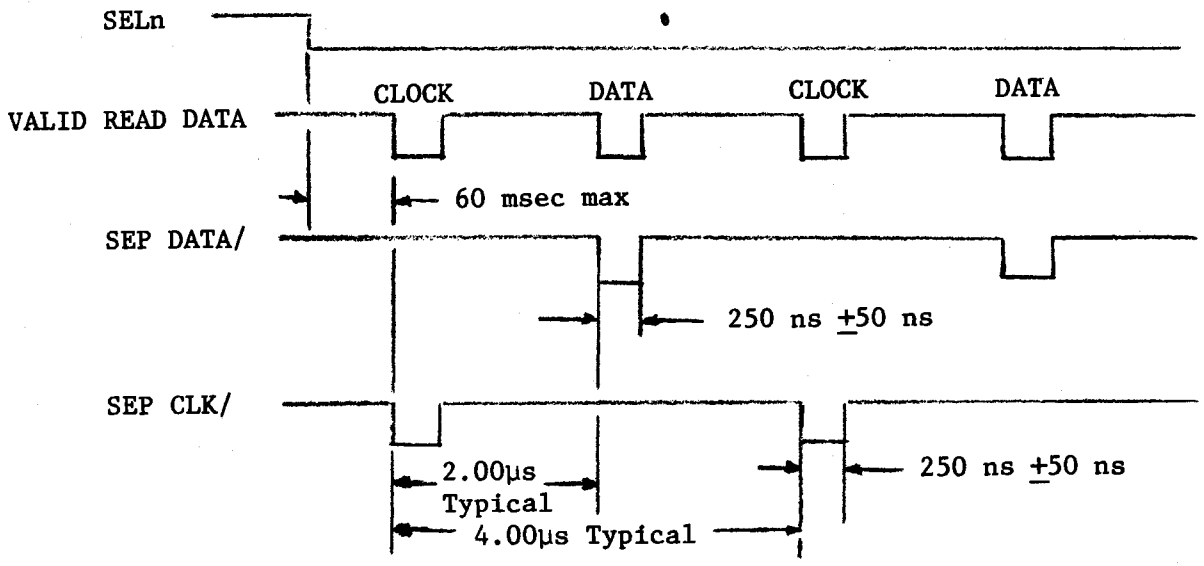


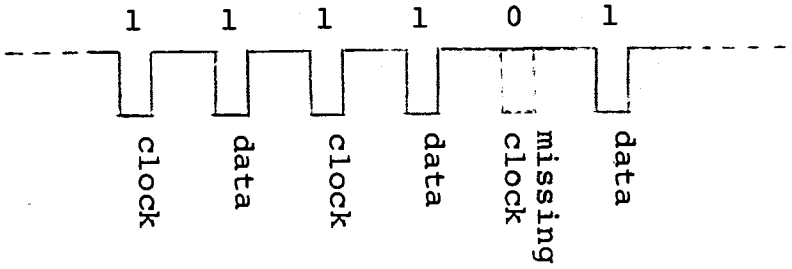
FIGURE 4-4

READ SIGNAL TIMING

clocks the 74191 counter at A7-14 and, on its trailing edge, clocks the data latch at A31 to the reset state.

Address marks differ from all other data bytes in that certain bits are missing. In fact, the third clock bit cell will be empty for all address marks. All of the clock bits are present in bytes other than address marks, and occur at approximately 4 µsec. intervals. The 9602 one-shot will produce a positive-going edge at its $\overline{Q}$ output (A4-7) 5 µsec. after it is triggered by a clock bit unless it is re-triggered by the next clock bit. When an address mark is being received, the missing clock bits will prevent the one-shot from being re-triggered; consequently, the positive-going edge from A4-7 will trigger another 9602 one-shot (A4-12) and will clock a 7474 latch at A15-11. The second one-shot will produce a short pulse (~73 nsec.) that will clock the latch at A31 with its trailing edge, in place of the missing clock bit (the SEP CLK line and this line from A4-10 are NORed at A37).

At this point let us pause and re-examine the data and clock bit patterns for an address mark. All address marks are preceded by at least 16 logical 0 data bits. The beginning bit pattern for all address marks is:



So when the first missing clock bit is detected, the Interface Board will have received at least 16 logical 0 data bits followed by the two logical 1 data bits shown above. The first logical 1 data bit received will already have been shifted into the data shift register on the Channel Board (see Section 3.2.5) and will be reflected by a true (low) level on the input data bit 0 line (IDO/). The second (most recent) logical 1 data bit will have pre-set the 7474 latch at A31-10. The high Q output of this latch

is applied to one input on the 7410 NAND gate at A16-13. ID0/ is inverted (to active high) and applied to one of the other NAND gate inputs. The final input to this 7410 gate is supplied by the "min/max" output from the 74191 counter. Because all address marks are preceded by at least 16 logical 0 data bits, the input data bit 1 line (ID1/) will not have been true (low) for at least 16 clock bits. Consequently, the 74191 counter will have counted up to 15 and activated its "min/max' output (A7-12) which disables further counting and provides the final high input to the NAND gate at A16-1. The 7410 gate will be activated, presenting a low level to the D-input of the latch at A15-12. When the 9602 one-shot times out because it has not received a clock bit for 5 μsec., the positive-going edge from its $\overline{Q}$ output (A4-7) will clock the 7474 latch at A15-11 to the reset state, causing a high level to appear on the latches $\overline{Q}$ output at A15-8. This $\overline{Q}$ output feeds the D-input on the out-of-phase latch. Approximately 73 nsec. later the second 9602 one-shot will timeout. The negative-going edge at its Q output (A4-10) will clock the out-of-phase latch (A15-3) to the set state.

When a clock bit is not present, the one-shot data separator in the diskette drive goes "out of phase" and sends the next data bit on the SEP CLK/ line. The Interface Board must compensate for this phenomenon (when the diskette drive is "out of phase" and when it is not). The out-of-phase latch (the second 7474 section at A15) will be in the set state (A15-5 will be true) when the drive is sending data bits on the SEP CLK/ line and clock bits on the SEP DATA/ line, and will be in the reset state (A15-6 will be true) when the drive is not "out-of-phase".

The three 7400 gates, the two 7410 gates and the 7408 gate shown on the schematic at A16, A17 and A47 are used to control the data strobe line (DATA SR STB), the data in line (SR DATA IN/), the clock strobe (CLK SR STB) and the clock in line (SR CLK IN/), depending on the state of the out-of-phase latch:

- The output at A16-6 produces one strobe for the clock shift register (CLK SR STB) during the address mark byte.

- The output at A47-11 produces compensated serial input to the data shift register (SR DATA IN/); if the drive is out-of-phase, SR DATA IN/ will be low (true); otherwise, SR DATA IN/ will reflect the level on SEP DATA/.

- The output at A17-6 produces the compensated serial input to the clock shift register (SR CLK IN/) by setting or resetting the latch at A39-12. If the drive is not out-of-phase, SR CLK IN/ will be true (low); otherwise, SR CLK IN/ will reflect the level on SEP DATA/.

- The output at A16-8 produces one strobe for the data shift register (DATA SR STB) during the address mark byte.

- The output at A17-3 produces CLK SR STB and DATA SR STB when a clock bit is received on SEP CLK/ during read operations.

- The output at A17-8 produces CLK SR STB and DATA SR STB during a disk write operation (see Section 4.2.3).

Timing for SR DATA IN/, SR CLK IN/ CLK SR STB and DATA SR STB is illustrated in Figure 4-5.

The bit counter portion of the synchronization block is responsible for determining when an entire byte has been serially received during read operations (i.e., when the data/clock shift registers are full), as well as determining when all eight bits of a particular byte have shifted out during write operations (i.e., when the shift registers are empty). Prior to a read or write operation, the microprogram, being executed on the Channel Board, generates the

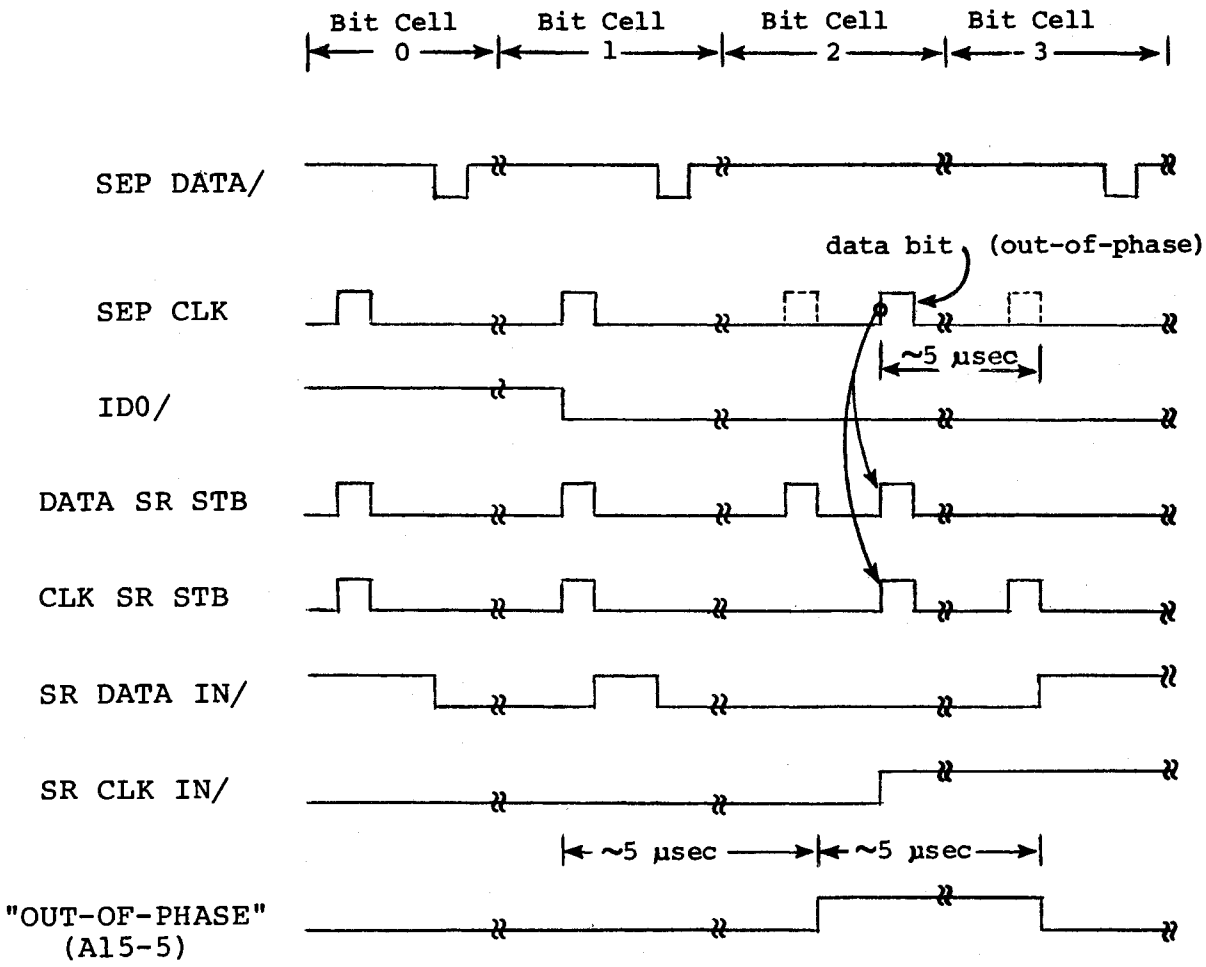


FIGURE 4-5

TIMING FOR ADDRESS MARK DETECTION

RSAMD control pulse (see Table 3-3) which resets the bit counter logic. The microprogram then generates the CSYNC control pulse which initializes the bit counter logic by presetting the 7474 latch at A18-10. The Interface Board will then be primed to search for an address mark to establish synchronization before it begins reading or writing data. When the beginning of the address mark is detected (i.e., when the out-of-phase latch is set), the 7474 latch at A18-3 is clocked set. When the next data strobe (DATA SR STB) is generated, the 7474 section at A25-11 is clocked set. The Q output from this latch disables the parallel load input on the 9316 counter (pin 9). The $\overline{Q}$ output from the latch feeds the P1 parallel input on the counter; P0 and P2 are held low while P3 is held high. Because the $\overline{Q}$ output had been high, the counter is pre-loaded to 10 (1010_2) when DATA SR STB goes true. This is because the beginning of an address mark is not detected until after two data bits had already been received. After six counts, the terminal count output (pin 15) on the counter goes high, activating the F line. This pre-sets the counter to 8 (1000_2) in anticipation of the next byte, and signals the microprogram that a full byte is present in the shift register.

During read operations, the reading of each byte is followed by pre-setting the counter to 8 and continuing. If the diskette controller is to perform a write operation after synchronizing with the diskette platter, the true (low) level on the DISK WRITE/ line will pre-set the latch at A25-10, forcing it to parallel load a value of 8 into the counter each time that F goes true.

4.2.3 WRITE CLOCK GENERATOR

The write clock generator block provides timing references for the writing of data and clock bits. This block consists of a 4 MHz crystal oscillator, two 74LS112 J-K flip-flops, three 74109 J-$\overline{K}$ flip-flops, a 7451 multiplexer, and assorted gating and inverting circuits, as shown on sheet 2 of the board schematic (Section 4.3).

The crystal oscillator provides the primary timing reference (4 MHz) that is divided by the two 74LS112 sections into a two-phase timing output (1 µsec. period, 50% duty cycle). These two-phase outputs are further divided by the three 74109 J-$\overline{K}$ flip-flops into three-phase timing outputs (4 µsec. period, 25% duty cycle). The two-phase outputs are ANDed with the three-phase outputs to produce three write clocks at A47-6, A1-8 and A1-6, as shown in Figure 4-6. The output from the 7408 gate at A47-6 is used in the synchronization block (Section 4.2.2) to produce the data and clock shift register strobes (DATA SR STB and CLK SR STB) during write operations. The output from the 7408 gates at A1-8 and A1-6 provide timing for the writing of data (or CRC) bits and clock bits, respectively. The data bits from the data shift register (on the Channel Board) or the CRC block (see Section 4.2.2) are gated through the 7451 multiplexer when the write data timing pulse (A1-8) is true. When the write clock timing pulse (A1-6) is true, the clock bits from the clock shift register (SR CLK OUT) are gated through the multiplexer. When a drive is selected, DISK WRITE produces the WRT GT/ signal (J1-27), and the output from the 7451 multiplexer is gated onto the WRITE DATA/ line (J1-3). Write data timing is shown in Figure 4-7.

Neither the WRITE DATA/ nor the WRT GT/ signals are true during disk read operations. Refer to Figure 4-4 for read timing.

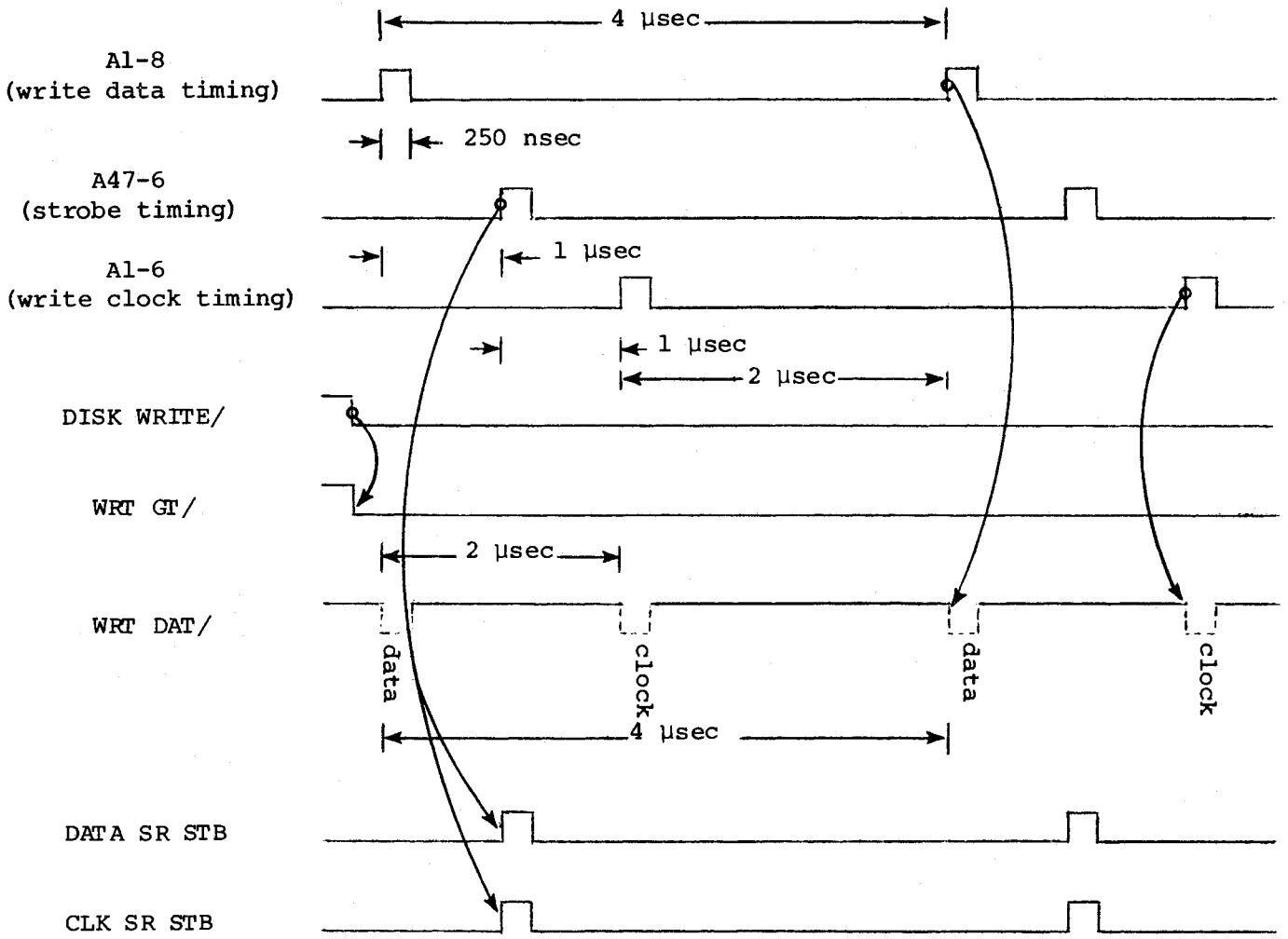


FIGURE 4-6
WRITE CLOCK TIMING

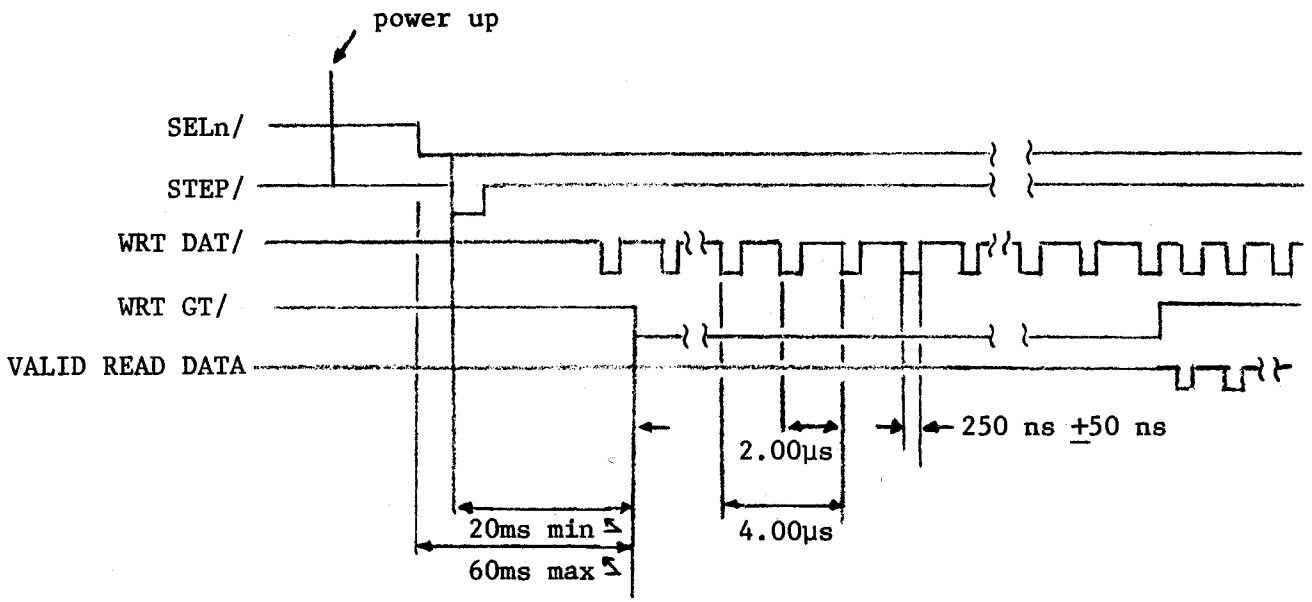


FIGURE 4-7

WRITE INITIATE TIMING

4.2.4 CYCLIC REDUNDANCY CHECK (CRC)

Two cyclic redundancy check (CRC) characters (16-bits) are generated for each data field (i.e., the address mark and 128 bytes of data) and are then appended to the data field as it is written to diskette. During all read operations, 16 CRC bits are generated as data is read; these CRC bits are then compared with the CRC bits that were appended to the field when it was written. CRC generation and checking are performed by the CRC block which consists of an 8503 Universal Polynomial Generator (UPG), a 7451 multiplexer and several inverting circuits, as shown on sheet 1 of the board schematic (see Section 4.3).

During write operations, data bits from the data shift register (on the Channel Board - see Section 3.2.5) are input to the 8503 UPG device (pin 11, SDI), as well as the 7451 multiplexer via the SR DATA OUT line. The SR OUT control level (maintained by the microprogram on the Channel Board - see Table 3-3) allows the data bits through the multiplexer and onto the WRT DAT/ line. The data bits, which are also being input to the 8503 device, cause the 8503 to generate the CRC characters (16 bits) for the 128 data bytes by "dividing" the data by the encoding polynomial ($x^{16} + x^{12} + x^5 + 1$). When the entire 129 bytes (1 address mark byte and 128 data bytes) have been written to disk, the microprogram removes the SR OUT pulse, which allows the 16 CRC bits being output by the 8503 device (pin 12, SDO) to pass through the 7451 multiplexer and be written onto the diskette immediately after the data.

During read operations, each data byte is shifted onto the 8503 UPG device as the succeeding data byte is being shifted into the data shift register (on the Channel Board). The data bits are carried on the SR DATA OUT line, just as during a write operation. The absence of the SR OUT control level (from the microprogram), however, prevents the data bits from being gated out onto the WRT DAT/ line. The 129 bytes are "divided" by the encoding polynomial to generate 16 CRC bits. After all 129 bytes have been read, the 16 CRC bits which were appended to the data when it was written are also shifted

into the 8503 device where they are compared with the CRC bits just generated. If the two sets of CRC bits match, the all zeroes output ($\overline{AZ}$, pin 13) goes true (low), and is applied to the D-input on a 7474 latch at A25-2. When the bit counter in address mark detection logic determines that all data has been shifted out of the data register (i.e., when the F signal goes true), the low level from the $\overline{AZ}$ output is clocked into the 7474 latch. The high $\overline{Q}$ output from this latch drives the AZ line (pin P2-41) to the Channel Board.

The microprogram controls the various operating modes of the 8503 UPG by maintaining the appropriate levels on the CRCMD control line (see Table 3-3). CRCMD, which feeds the shift right ($\overline{SR}$) input to the UPG, is usually low, causing logical one's to be shifted through the UPG (via its D1 input). It is only when CRC characters are being generated or checked that CRCMD presents a false (high) level to the active-low $\overline{SR}$ input. The 8503 device is clocked by the data shift register strobe signal (DATA SR STB).

4.2.5 BUS CONTROL

The bus control block maintains the diskette controller interface with the SBC system bus. This block consists of seven 7474 D-type flip-flops, two 74H74 D-type flip-flops, three 74109 J-$\overline{K}$ flip-flops, a 9602 one-shot multivibrator and assorted gating and inverting circuits, as shown on sheet 3 of the board schematic (see Section 4.3).

Before the diskette controller can transfer data to or from system memory, the bus control block must request and be granted master control of the SBC system bus. When the Diskette Channel requires access to memory, the microprogram (being executed on the Channel Board - see Chapter 3) will initiate the bus request sequence by generating the SMREQ control pulse (see Table 3-3). SMREQ will cause the 7474 latch at A40-4 to be pre-set unless the inhibit

memory write latch (A48-5) is set.

The inhibit memory write latch will be set during VERIFY CRC diskette operations, in which data is read and verified but is not transferred to memory. The inhibit memory write latch is set and reset by the SINH and RINH control pulses, respectively (both are generated by the microprogram, see Table 3-3).

The Q output from the 7474 latch which is pre-set by SMREQ (A40-5) is applied to the D-input on the transfer request latch (at A40-12). The next bus clock pulse (BCLK/) will clock this latch to the set state. The low $\overline{Q}$ output drives the XFER REQ/ line (pin P2-39) which informs the SBC system that the diskette controller requires use of the system bus. The Q output from the transfer request latch feeds the J-input of the data overrun (DOR/) latch and the D-input of the bus request (BREQ/) latches.

If the microprogram requests a memory access again (i.e., if SMREQ is generated again) before the current access is completed, the data overrun latch will be clocked to the set state, and its $\overline{Q}$ output will drive a true (low) level on the DOR/ error line to the Channel Board (pin P2-16).

The bus request latch will be clocked set by the next bus clock pulse (BCLK/), causing BREQ/ (pin P1-18) to go true (low). BREQ/, like XFER REQ/, requests use of the SBC bus. XFER REQ/, however, precedes BREQ/ by one BCLK/ pulse.

If no other master module is using the bus (i.e., if BUSY/ is false), and if no higher priority module is requesting the bus (i.e., the bus priority in line, BPRN/, is true), the next bus clock pulse (BCLK/) after BREQ/ goes true will clock the busy latch set. The $\overline{Q}$ output of the busy latch enables the 8093 circuit which drives BUSY/ (pin P1-17). BUSY/ informs the SBC system that the diskette controller has master control of the bus. The Q output of the busy latch enables the SELECTED signal (pin P2-43). The Q

output of the busy latch is also applied to the D-input of a 7474 latch at A33-2. The BCLK/ pulse after BUSY/ goes true will clock this 7474 latch set. The low $\overline{Q}$ output, in turn, enables the 8097 circuits which drive the memory read (MRDC/) and memory write (MWTC/) commands at pins P1-19 and P1-20, respectively.

The microprogram maintains the MEM WT control level (see Table 3-3), which indicates whether memory is to be read from or written into. MEM WT is applied to two 7400 NAND gates (at A34) in its true and complemented form. The other input to these two 7400 gates is supplied by the Q output of the transfer request latch. The output from one of these gates will be true, and will constitute the memory read (MRCD/) or write (MWTC/) command. In addition, the read or write signal will, in conjunction with the SELECTED signal (mentioned above), clock a 74H74 latch at A46-1 to the set state. This latch prevents the bus control block from relinquishing master control of the system bus before the current memory read or write operation is completed by maintaining a high level at the $\overline{K}$ input to the busy latch (a 74109 J-$\overline{K}$ flip-flop). This high at the $\overline{K}$ input prevents the latch from being reset. Only after the memory access is completed will the absence of a memory read or write signal clear the 74H74 latch and allow the busy latch to be reset.

Recall that while logic on the Interface Board requests control of the bus, then generates the memory read or write command, it is the Channel Board which actually drives the system data and address busses (refer to Chapter 3).

When the memory has accepted the data to be written or output the data which was read, it generates a transfer acknowledge signal (XACK/) which is received at pin P1-23 on the Interface Board. XACK/ triggers a 9602 one-shot. The high pulse (~ 73 nsec. wide) at the Q output of the one-shot drives the STB MEM IN line (pin P2-29) which allows the Channel Board to accept data read from memory.

The $\overline{Q}$ output from the one-shot clocks the 7474 latch at A40-3 to the reset state. As a result, succeeding bus clock pulses (BCLK/) can reset the bus control logic, and cause the Interface Board to relinquish master bus control to another module.

Bus control timing is illustrated in Figure 4-8.

In addition to being the master module during memory access operations, the diskette controller also acts as the "slave" module during I/O operations in which the CPU executes a channel command for the diskette controller. The bus control block accepts the I/O read (IORC/) or I/O write (IOWC/) command from the CPU, inverts it and passes it to the Channel Board on the READ CMD line (pin P2-60) or the WRT CMD line (pin P2-53).

The "slave" module must acknowledge all commands from the master. The bus control block performs this function in response to channel commands. Receipt of a "stop", "reset", "read subsystem status" or "read result type" channel command (see Chapter 2) will clock the acknowledge latch (at A48-11) to the set state. The acknowledge latch is set at the proper time by the SACK control pulse from the microprogram for each of the other channel commands. The $\overline{Q}$ output from the acknowledge latch feeds an 8093 inverting driver which is enabled by the ENABLE signal from the Channel Board (pin P2-32) and either the I/O read (IORC/) or I/O write (IOWC/) signal. The output from the 8093 circuit (XACK/) is driven through pin P1-23. XACK/ is reset when IORC/ or IOWC/ go false.

The bus control block also has an 8 position rotary switch which connects the interrupt line (INT/) from the Channel Board (pin P2-40) with one of the eight system priority interrupt request lines, INTØ/ - INT7/ (via a 7409 gate which provides the required electrical characteristics for the MDS bus).

In addition, the two phase clock pulses (CLK1/ and CLK2/) are generated in the bus control block. CLK1/ and CLK2/ are derived from the SBC common clock, CCLK/ (9.8 MHz). CCLK/ is divided by the two 74109 J-$\overline{K}$ flip-flops at A50 to produce CLK1/ and CLK2/ (201.2 nsec. period, 25% duty cycle) at the outputs of two 7410 NAND gates (A51-6 and A51-12), as shown in Figure 4-9.

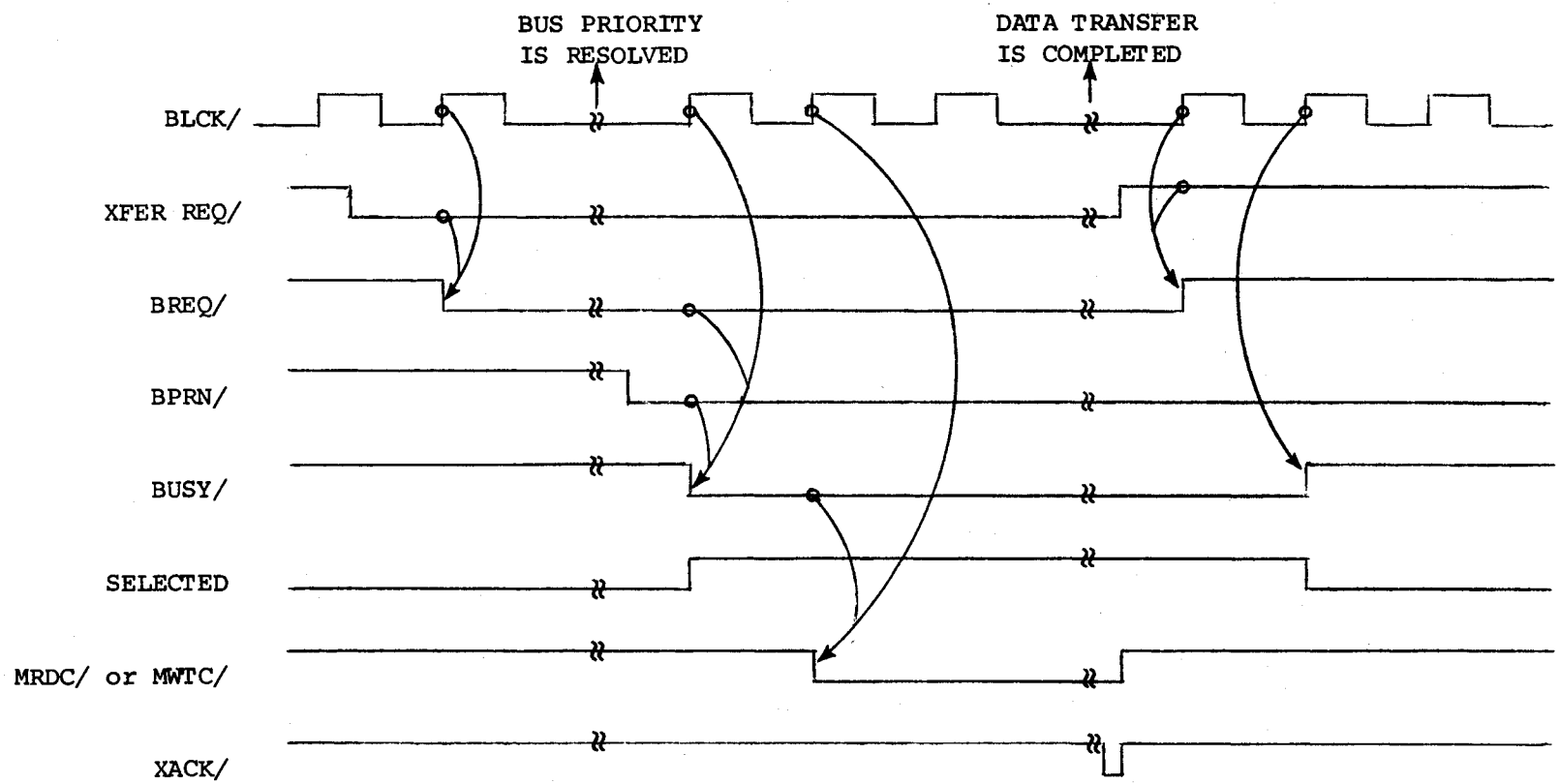


FIGURE 4-8
BUS CONTROL TIMING

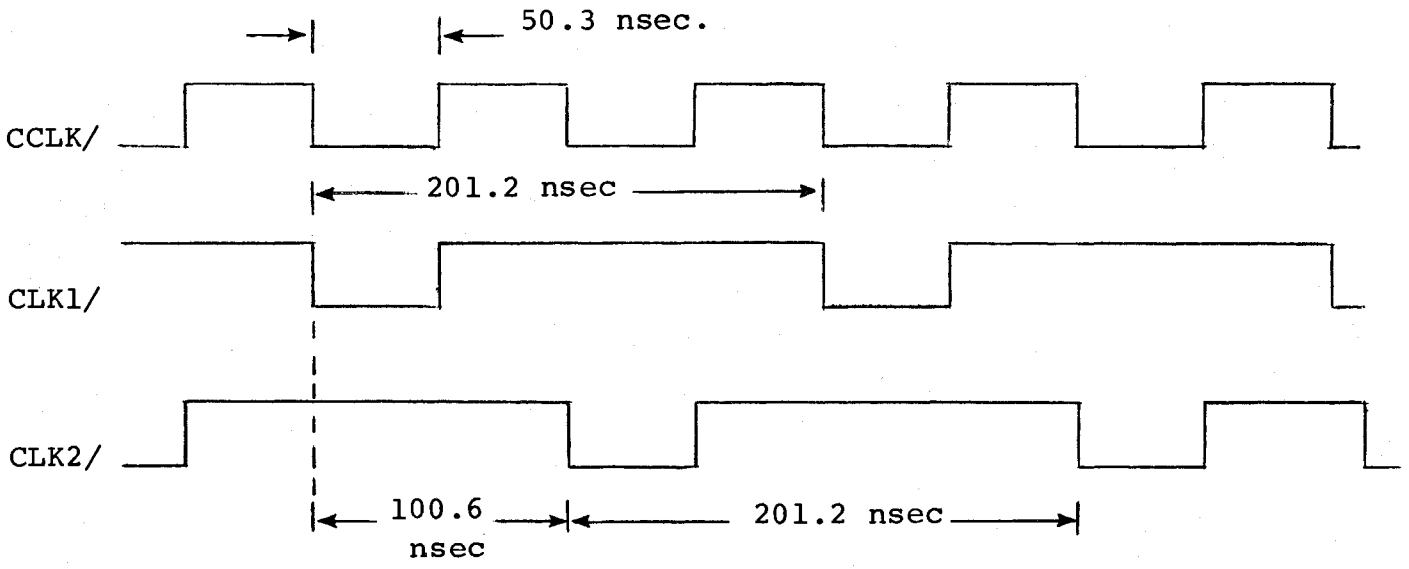


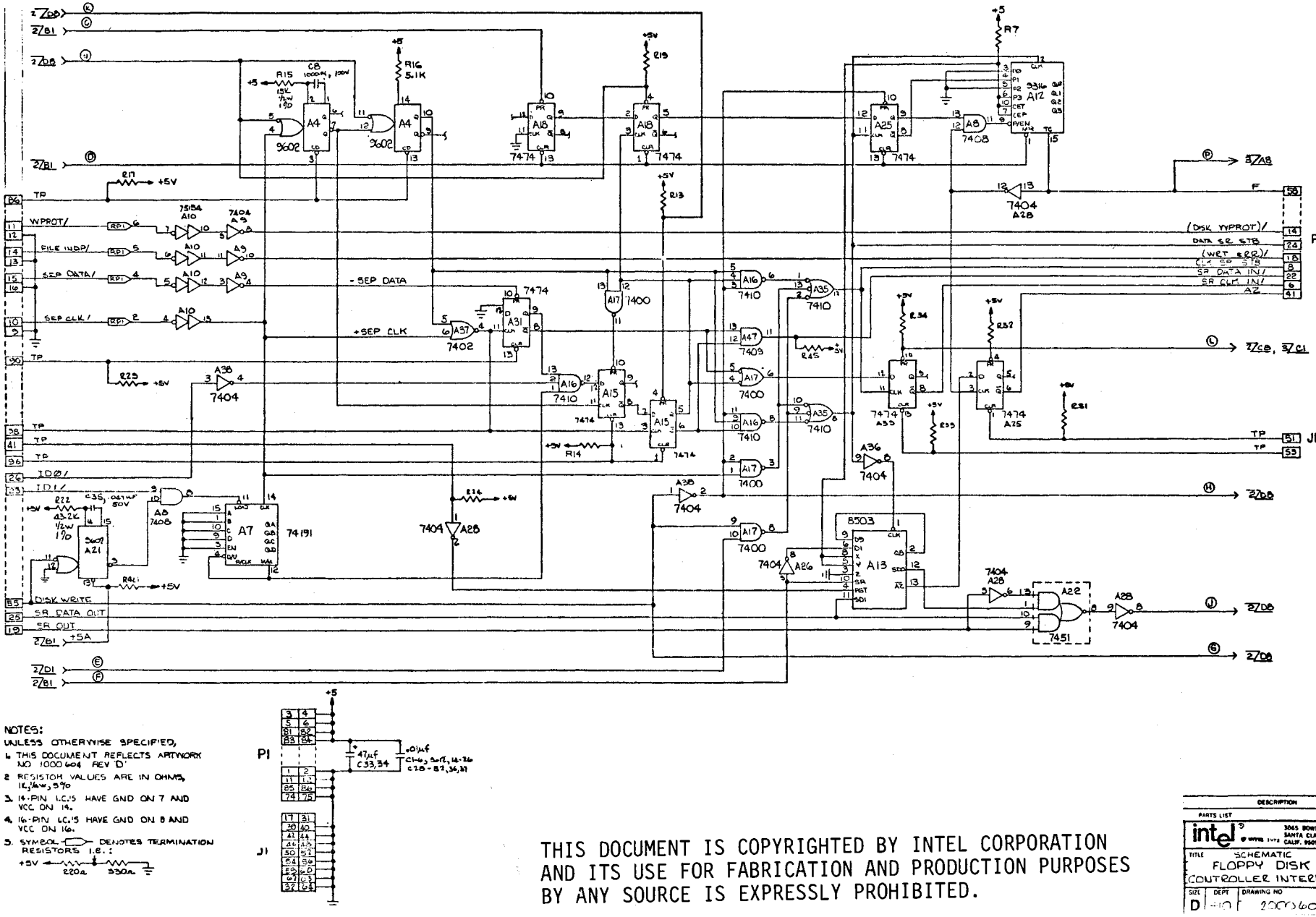
FIGURE 4-9

CLK1/ AND CLK2/ TIMING

4.3 SCHEMATICS/PIN LISTS: INTERFACE BOARD

The schematic drawing (3 sheets) for the Interface Board is provided in Figure 4-10. These schematics reflect the latest revision level of the board. Appendix A contains a copy of the assembly drawing for the board; the assembly drawing should be consulted for the location of the various devices on the board.

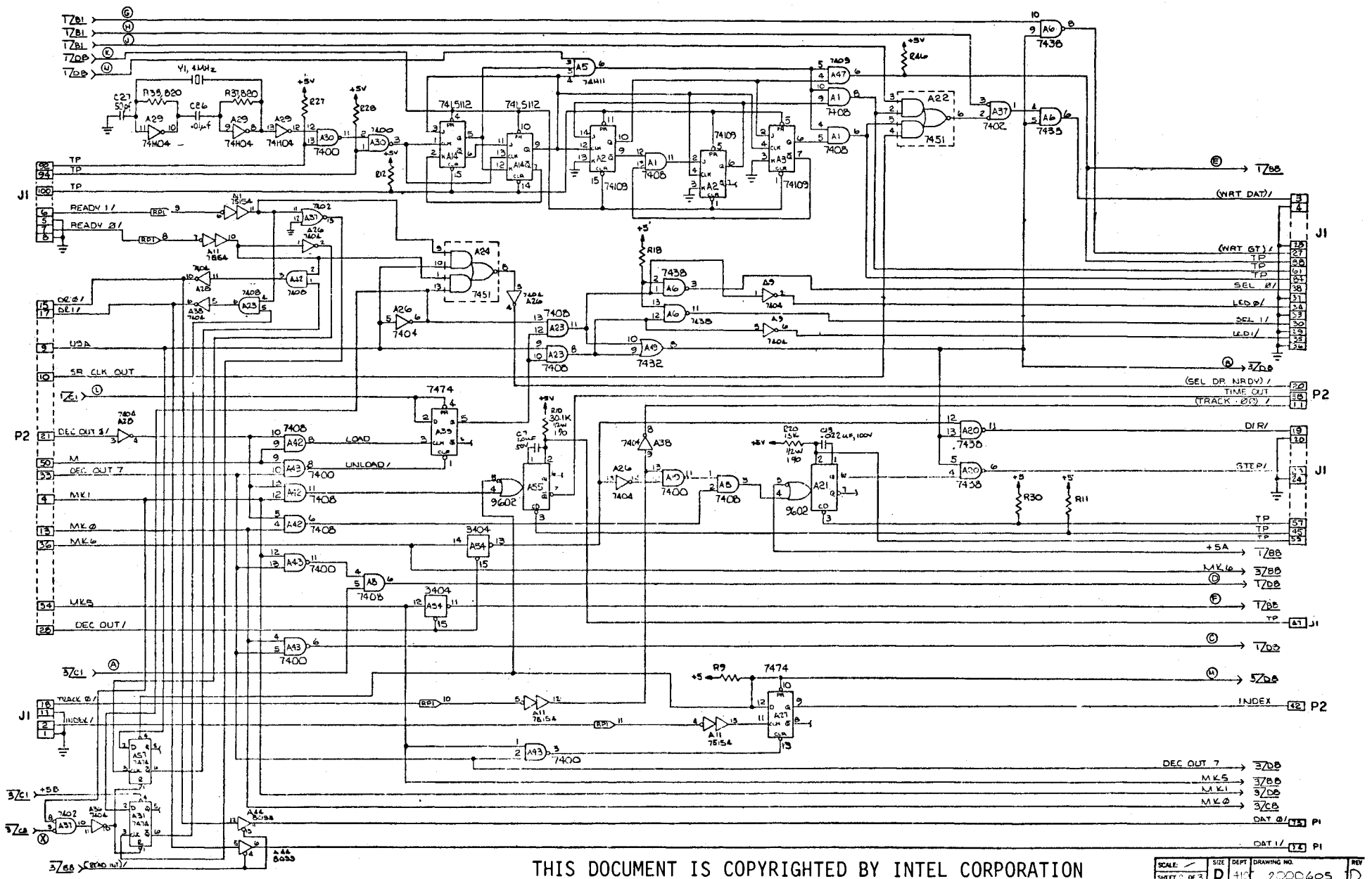
Table 4-1 lists the pins and designated signal functions for the 86-pin P1 bus connector. Table 4-1 lists the same information for the 60-pin P2 controller connector, while Table 4-3 provides this information for the 100-pin J1 drive connector.



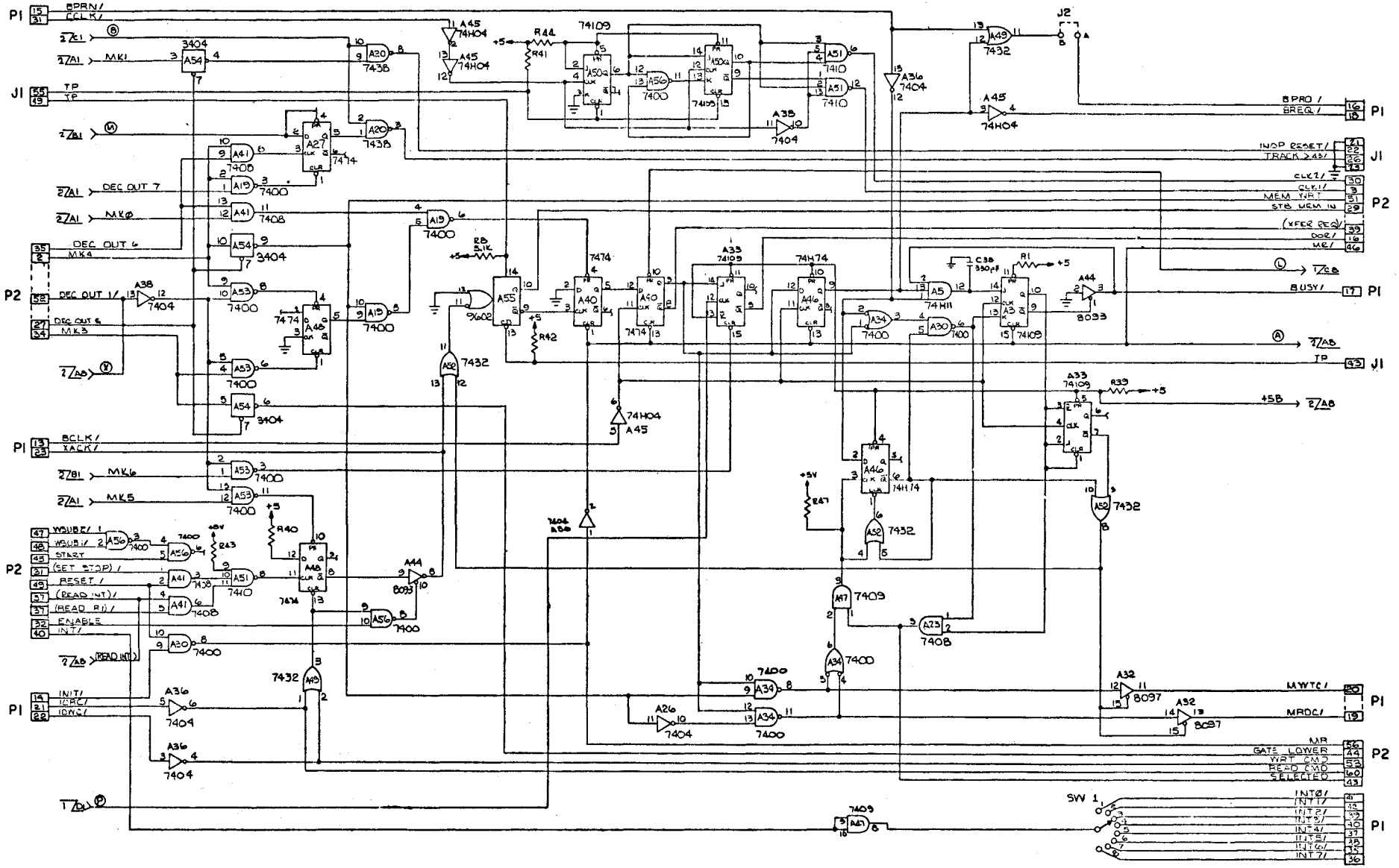
NOTES:
UNLESS OTHERWISE SPECIFIED,

1. THIS DOCUMENT REFLECTS ARTWORK NO 1000 604 REV 'D'
2. RESISTOR VALUES ARE IN OHMS, 1/4W, 5%
3. 14-PIN I.C.'S HAVE GND ON 7 AND VCC ON 14.
4. 16-PIN I.C.'S HAVE GND ON 8 AND VCC ON 16.
5. SYMBOL ⊐ DENOTES TERMINATION RESISTORS I.E.: +5V — 220Ω — 330Ω

DESCRIPTION	
PARTS LIST	
intel	3065 BOWERS AVE. SANTA CLARA CALIF. 95051
TITLE	SCHEMATIC FLOPPY DISK CONTROLLER INTERFACE
SIZE	D
DEPT	410
DRAWING NO	2000605
REV	D



SCALE:	SIZE	DEPT	DRAWING NO.	REV
SHEET 2 OF 3	D	410	2000605	D



SCALE:	SIZE	DEPT	DRAWING NO.	REV
SHEET 3 OF 3	D	RD	2000605	D

TABLE 4-1

PIN LIST: P1 BUS CONNECTOR

PIN	SIGNAL	FUNCTION
1 ↓ 12		
13	BCLK/	Bus clock (9.803 MHz)
14	INIT/	System initialization
15	BPRN/	Bus priority in
16	BPRO/	Bus priority out
17	BUSY/	Bus busy
18	BREQ/	Bus request
19	MRDC/	Memory read command
20	MWTC/	Memory write command
21	IORC/	I/O read command
22	IOWC/	I/O write command
23	XACK/	Transfer acknowledge
24 ↓ 30		
31	CCLK/	Common clock (9.803 MHz)
32		
33		
34		
35	INT6/	Priority interrupt request lines
36	INT7/	
37	INT4/	
38	INT5/	
39	INT2/	
40	INT3/	
41	INT0/	
42	INT1/	
43 ↓ 86		

TABLE 4-2
PIN LIST: P2 CONTROLLER CONNECTOR

PIN	SIGNAL	FUNCTION
1		
2	MK4	Mask bit 4
3	CLK1/	Diskette controller clock 1
4	MK1	Mask bit 1
5		
6	SR CLK IN/	Serial clock in line
7		
8	CLK SR STB	Serial clock strobe
9	USA	Unit select bit
10	SR CLK OUT	Serial clock out line
11	TRACK00/	Track 00 detected
12		
13	MK0	Mask bit 0
14	DISK WPROT/	Disk write protected
15	DR0/	Drive 0 ready
16	DOR	Data overrun error
17	DR1/	Drive 1 ready
18	WRT ERR/	Write error
19	SR OUT	Shift register out control level
20	SEL DR NRDY/	Selected drive not ready
21	DEC OUT 0/	Control decoder output 0
22	SR DATA IN/	Serial data in line
23	ID1/	Input data bit 1
24	DATA SR STB/	Serial data strobe
25	SR DATA OUT	Serial data out line
26	ID0/	Input data bit 0
27	DEC OUT 5/	Control decoder 5 output
28	DEC OUT 4/	Control decoder 4 output
29	STB MEM IN	Strobe memory data in
30	CLK2/	Diskette controller clock 2

Table 4-2 (Continued)

PIN	SIGNAL	FUNCTION
31	SET STOP/	"Stop diskette" channel command
32	ENABLE	Diskette controller addressed
33	DEC OUT 7	Control decoder output 7
34	MK3	Mask bit 3
35	DEC OUT 6	Control decoder output 6
36	MK6	Mask bit 6
37	RD RI/	"Read Result Type" channel command
38	TIME OUT	10 msec. timing pulse
39	XFER REQ/	Controller requests SBC bus
40	INT/	Interrupt line
41	AZ	All zeros, valid CRC check
42	INDEX	Index mark detected
43	SELECTED	Controller has control of SBC bus
44	GATE LOWER	Input low order data byte
45	START	Microprogram responding to channel command
46	MR/	Master reset
47	WSUB2/	Not used at present
48	WSUBl/	Not used at present
49	RESET/	"Reset" channel command
50	MK2	Mask bit 2
51	MEM WRT	Write data to SBC memory
52	DEC OUT 1/	Control decoder output 1
53	WRT CMD	I/O write command
54	MK5	Mask bits
55	WRTGT	Write gate control level
56	MR	Master reset
57	RD INT/	"Read subsystem status" channel command
58	F	Shift registers full or empty
59		
60	RD CMD	I/O read command

TABLE 4-3
J1 DRIVE CONNECTOR

PIN	SIGNAL	FUNCTION
1	Gnd	Ground
2	INDEX/	Index hole indicator
3	WRT DAT/	Write data line
4	Gnd	Ground
5	Gnd	Ground
6	READY1/	Ready line from drive #1
7	READY0/	Ready line from drive #0
8	Gnd	Ground
9	Gnd	Ground
10	SEP CLK/	Clock input line
11	WPROT/	Write protect indicator
12	Gnd	Ground
13	Gnd	Ground
14	FILE INOP/	File inoperable
15	SEP DATA/	Data input line
16	Gnd	Ground
17	Gnd	Ground
18	TRACK0/	Track 0 indicator
19	DIR/	Direction indicator
20	Gnd	Ground
21	Gnd	Ground
22	IN OP RESET	File inoperable reset
23	STEP/	Steps head one track
24	Gnd	Ground
25	Gnd	Ground
26	TRACK>43/	Track greater than 43 line
27	WRT GT/	Write gate line
28	Gnd	Ground
29	Gnd	Ground
30	SEL1/	Select/load head drive #1
31	Gnd	Ground
32	Gnd	Ground
33	Gnd	Ground
34	LEDØ/	Drive #0 selected indicator
35	LED1/	Drive #1 selected indicator
36	Gnd	Ground
37	Gnd	Ground
38	SEL0/	Select/load head drive #0
39	Gnd	Ground
40	Gnd	Ground
41	TP	Test point
42	Gnd	Ground
43	TP	Test point
44	Gnd	Ground
45	TP	Test point
46	Gnd	Ground
47	TP	Test point
48	Gnd	Ground
49	TP	Test point
50	Gnd	Ground

Table 4-3 (Continued)

PIN	SIGNAL	FUNCTION
51	TP	Test point
52	Gnd	Ground
53	TP	Test point
54	Gnd	Ground
55	TP	Test point
56	Gnd	Ground
57	TP	Test point
58	Gnd	Ground
59	TP	Test point
60	Gnd	Ground
61		
62	Gnd	Ground
63	Gnd	Ground
64	Gnd	Ground
65		
66		
67		
68		
69		
70		
71		
72		
73		
74		
75		
76		
77		
78		
79		
80		
81		
82		
83		
84	TP	Test point
85		
86	TP	Test point
87		
88	TP	Test point
89		
90	TP	Test point
91		
92	TP	Test point
93		
94	TP	Test point
95		
96	TP	Test point
97		
98	TP	Test point
99		
100	TP	Test point

CHAPTER 5
THE DISKETTE DRIVES

The Diskette Controller can operate a CDC 9404 or Shugart 800/800R Flexible Disk Drive (FDD) or other electrically comparable drives. The FDD is a reliable random access storage device utilizing a single, removable diskette as the storage medium. This chapter summarizes the manufacturers information on the drive. The FDD uses an IBM 3740 media compatible diskette. The disk itself is 7.88 inches in diameter and is contained in an 8 inch square protective envelope. To load the disk, the entire envelope is inserted into a slot in the FDD behind a small access cover.

Rotating at 360 rpm, data is read or written on the 77 tracks of the single recording surface at a rate of 249,984 bits per second in single density mode to provide a total unformatted capacity of 3-megabits. The single head need only be in contact with the media during actual data transfer operations. Track accessing is accomplished using a stepping motor. Index is detected using a photo-optical technique to sense the physical index hole in the disk cartridge.

The FDD contains all the analog read, write and control electronics necessary to perform data transfer operations using only simple control commands.

5.1 FUNCTIONAL DESCRIPTION

The following paragraphs describe the major components of the FDD.

Electronics:

All electronic circuitry required to convert from the digital I/O to the analog read/write and positioning information is contained on a single circuit board mounted underneath the deck. Logic is T^2L with a minimal number of discrete analog components.

Positioner:

Positioning of the read/write head is accomplished using a stepping motor driving a lead screw shaft.

Head Loading Mechanism:

Head loading is achieved by a solenoid/pressure pad scheme. When the solenoid is activated, it releases the pressure pad arm which, in turn, pushes the disk against the read/write head.

Spindle Motor:

The spindle is belt driven from the spindle motor, which maintains a speed of 360 RPM $\pm$ 3.5%.

Disk Loading Mechanism:

Disk loading is accomplished when the disk loading access cover is closed. A two piece expandable centering cone is loaded against the center of the disk on the side opposing the spindle, causing the disk to be located accurately between the centering cone and the spindle. The centering cone, disk, and spindle then rotate together.

Head:

The read/write head is a single gap head plus track edge erase gaps. Nominal track width is 0.013 inches. The read/write to erase gap spacing is 0.035 inches.

Mechanical Framework:

A solid deck die casting provides a clean construction to enhance reliable operation. The deck casting provides for

spindle housing, disk cartridge registration and envelope stabilization surfaces, drive motor mounting surface, stepper motor mounting surface, electronic circuit board mounting as well as providing a stable base for integration and operation of unit components. A secondary frame die casting provides stable mounting, in relationship to the main deck casting, for the disk centering cone and loading mechanism.

A simple disk loading access cover completes the disk loading/unloading mechanism and prevents the possibility of objects falling into the unit.

Diskette:

Single disk cartridge in a sealed envelope. The envelope size is 8" x 8" with a recording disk diameter of 7.88". The FDD uses the IBM 2305830 type diskette.

The disk has 77 concentric recording tracks spaced .02083 inch apart on a single surface. Track centerlines are calculated by the formula: centerline radius = $2.029 + \frac{(76-N)}{48}$. N is the physical track number.

5.2 PERFORMANCE CHARACTERISTICS

The following sections list the performance characteristics of the FDD.

5.2.1 Recording Characteristics

Double Frequency, self-clocking, serial by bit for single density operation, providing:

Outer track (TR00) - 1836 bits/inch
Inner track (TR76) - 3268 bits/inch
Track Density - 48 tracks/inch

5.2.2 Bit Transfer Rate

Based on a nominal disk speed of 360 rpm, the bit transfer rate is 249,984 bits per second single density.

5.2.3 Data Capacity

The data capacity listed below is an unformatted maximum and will be reduced by formatting allowances for sector operation and spare track allocation.

	Single Density
Track capacity	41,664 bits
Disk capacity	3,208,128 bits

5.2.4 Latency Time

Latency time is the time required to reach a particular point on a track after positioning is complete. It is a function of spindle speed.

The spindle speed for the 9404 FDD is 360 RPM $\pm$ 3.5%. The speed tolerance includes motor performance, pulley tolerance, spindle variation, $\pm$ 10% AC line voltage variation, and a $\pm$ 2% AC line frequency variation.

Based on a nominal disk speed of 360 rpm, the average latency time is 83.33 milliseconds.

Based on a minimum disk speed of 347 rpm (360 - 3.5%) the maximum latency time is 173 milliseconds.

5.2.5 Positioning Characteristics

The time for a single track move is 20ms including settling time. This is defined as the time to move between any pair of adjacent tracks. Multiple track moves can be made at 10ms per step plus 10ms settling time.

The random average positioning time is 260ms including settling time. This is defined as the summation of the move times for all possible moves divided by the number of possible moves.

The maximum positioning time is 770ms. This is defined as the time to move the head from track 00 to track 76 or from 76 to 00 and includes settling time.

5.2.6 FDD Start and Stop Time

The FDD spindle runs at all times while power is applied to the unit and is not stopped for disk loading or unloading. Since the time for the disk to reach the same speed as the spindle is negligible (less than 2 seconds) there is essentially no operator waiting time required.

5.2.7 Error Recovery

Read/Write Errors

To guard against degradation from imperfections in the media, it is recommended that no more than 4 attempts to write a record be used when read after write errors are encountered. In the event a record cannot be successfully written within 4 attempts, it is recommended that the sector or track be labeled defective and an alternate sector or track assigned. If more than 2 defective tracks are encountered, it is recommended that the disk be replaced.

In the event of a read error up to 10 attempts should be made to recover with re-reads. If after 10 attempts the data has not been recovered, step the head several tracks away and re-position to recover the data. If the error persists, the sequence should be attempted at least 10 times. Unloading the head when data transfers are not imminent will increase the data reliability and extend the disk life.

Seek Errors:

Seek errors will rarely occur unless the stepping rate is exceeded. In the event of a seek error, recalibration of track location can be achieved by repetitive Step Out commands until a track 00 signal is received.

5.2.8 Environmental Limits

Temperature and Humidity:

The FDD can withstand the following conditions, if the combined rate of temperature and humidity change precludes condensation of moisture on any part of the unit.

	Operating	Storage and Transit
Temperature, ^{o}F	50^{o} to 100^{o}	-30^{o} to 150^{o}
Temperature, ^{o}C	10^{o} to 38^{o}	-35^{o} to 65^{o}
Temp. Change $^{o}F/Hr.$	12	60
Temp. Change $^{o}C/Hr$	6.7	33
Relative Humidity	20% to 80%	5% to 95%
Max. Wet Bulb Temp. ^{o}F	80	
Max. Wet Bulb Temp. ^{o}C	26.7	

5.2.9 Write Protect

Write protect uses a photo-optical sensor that senses the presence of a write protect hole in the diskette jacket (see Figure 5-4). This feature prohibits the controller from accidently writing on a protected disk. The diskette can be written upon, if so desired, by masking the write protect hole.

5.3 INTERFACE SPECIFICATION

The following paragraphs describe the control and data interface lines, shown in Figure 5-1.

All signal lines must be terminated at the receiver with an impedance of 130 ohms, nominal. The following definitions will be used in the signal definitions:

Logic Level	Status Signals	Command Signals
High (False) (Ø)	2.4V min.	2.0V min.
Low (True) (1)	0.4V max.	0.8V max.

Figure 5-2 illustrates the basic driver and receiver circuits for the PDD.

Control and Data Lines to the FDD:

1. Step

   A 10 microsecond (minimum) logic 1 level pulse on this line causes the head to move one track inward or outward from the center of the disk, depending upon the state of the direction line.

2. Direction

   A logic 1 level on this line causes the head to move toward the center of the spindle when a step signal occurs. This line must be true before and after step.

3. Unit Select 1,2

   Two unit select lines are used by the controller to select the appropriate drive. These two lines are mutually

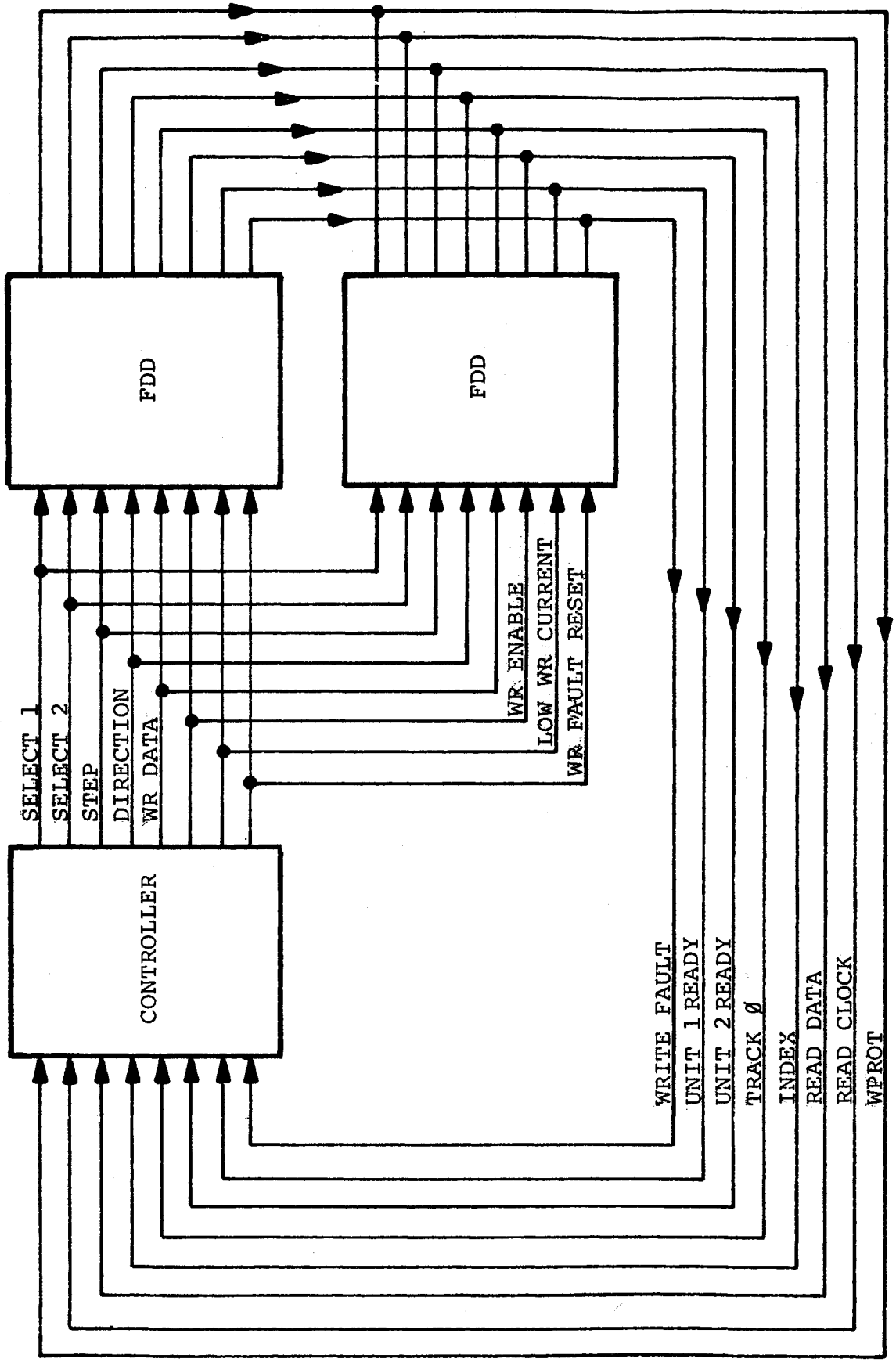
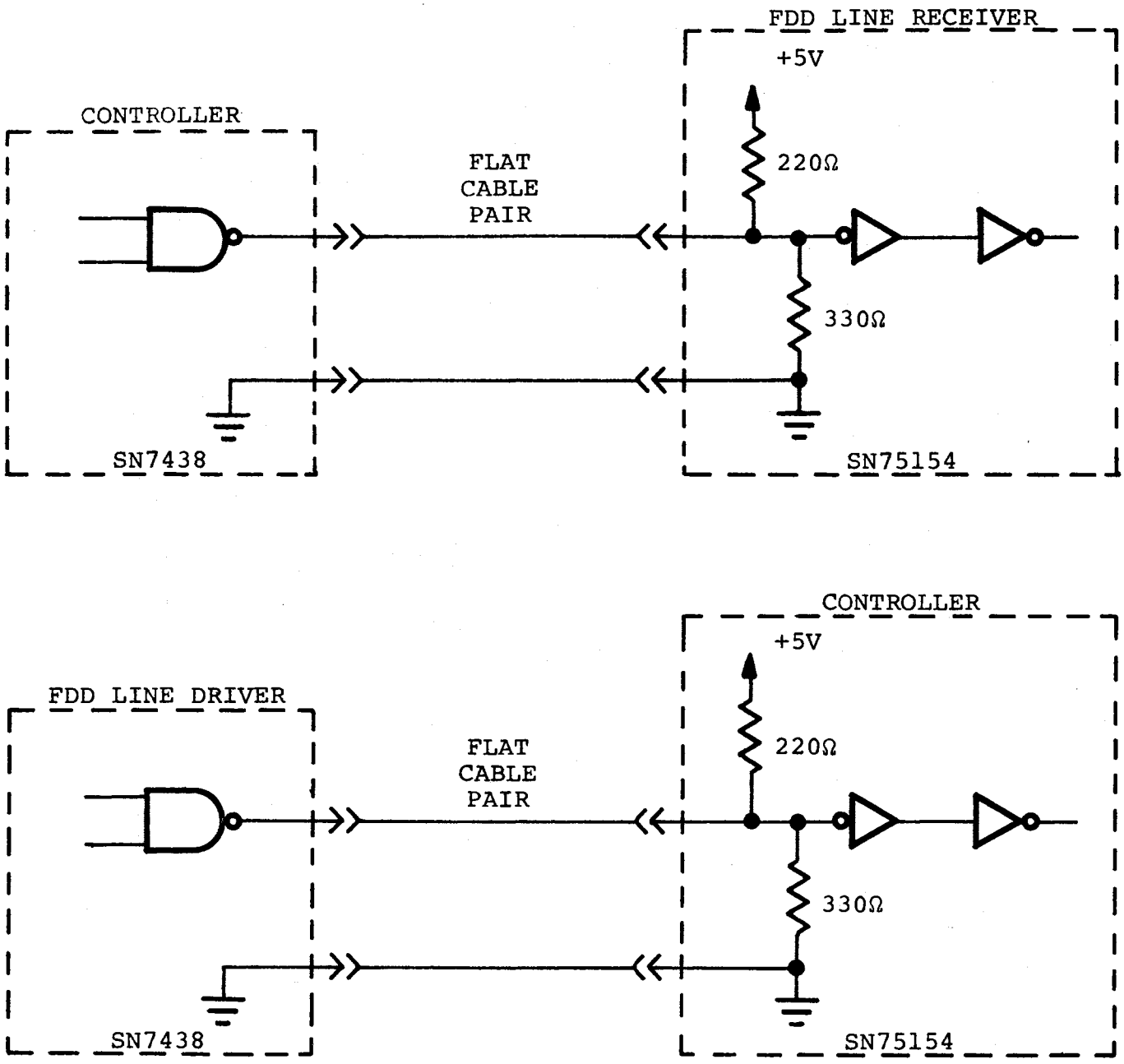


FIGURE 5-1

FDD/FDCC INTERFACE LINES



DRIVER

FIGURE 5-2

FDD DRIVER/RECEIVER CIRCUITS

exclusive; only one drive can be selected at a time. The lines must remain active during any command from the controller. Logical Unit assignment is accomplished on the drive Printed Circuit Board.

A jumper option on the disk PCB is used to maintain a logic 1 level on the head load line. This causes the head to be loaded automatically once a unit is selected. Whenever an operation is NOT pending, the unit is deselected, thus unloading the head. This increases head and media life. See Section 5.5 for more specific information.

4. Unit Ready Interrupt 1,2

There are two Unit Ready Interrupt lines, one from each unit, indicating that the unit power is on, the door is closed and the diskette has reached 70% of its final speed. The Unit Ready Interrupt assignments are accomplished on the drive Printed Circuit Board. See Section 5.5 for more specific information.

5. Write Enable

To enable the FDD write driver, this line is held at a logic 1.

To disable the FDD write driver and enable the FDD read circuitry, this line is held at logic 0.

6. Write Fault Reset (CDC only)

A logic 1 level on this line clears the Write Fault Latch.

7. Write Data (Refer to Figure 5-3)

This line contains the composite double frequency coded Write Clock and Data information to the FDD. The Write Clock and Data Pulses must be 250 nanoseconds ± 20% in length and are true at the logic 1 level. Information to be recorded on the disk is derived from the trailing edge of each pulse (i.e., at the logic 1 to logic 0 transition point).

8. Low Current (CDC only)

A logic 1 level on this line causes reduced write current to be used during a write operation. Low current should be true for all tracks greater than 41 (inner tracks). It is not necessary to gate low current with write enable in the controller.

9. Write Protect

This line will be a logical 1 when the write protect hole in a diskette jacket is present. When logical 1 this line will disable write and erase current circuitry. (For hole location, refer to Figure 5-4).

Control and Data Lines from the FDD:

1. Index

This line gives an indication of the relative position of the disk by outputting a logic 1 pulse for every revolution of the risk. The pulse of 1.5 ± 0.7 milliseconds is generated by sensing the index hole in the disk using a photo-optical technique.

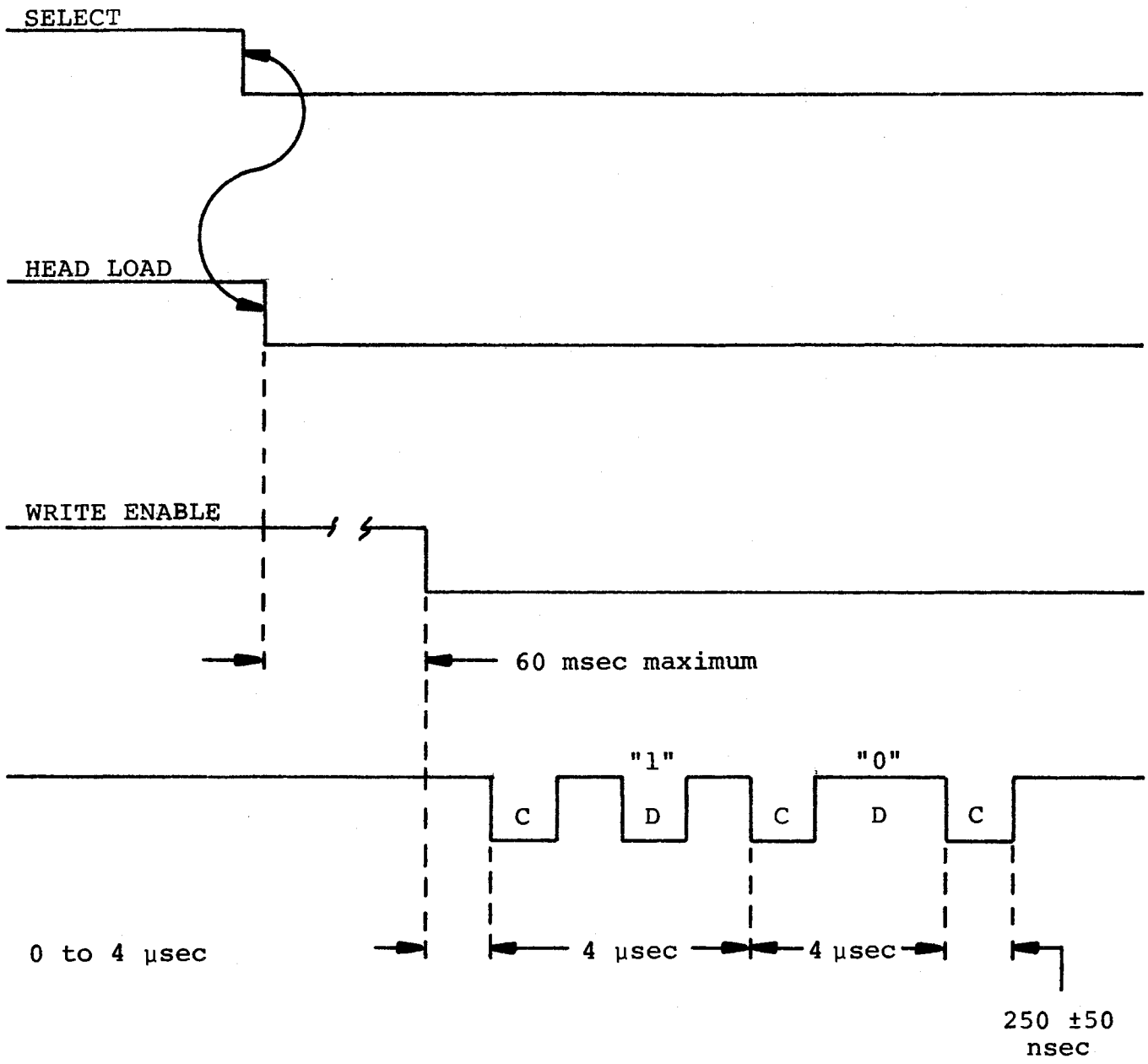


FIGURE 5-3

WRITE DATA TIMING

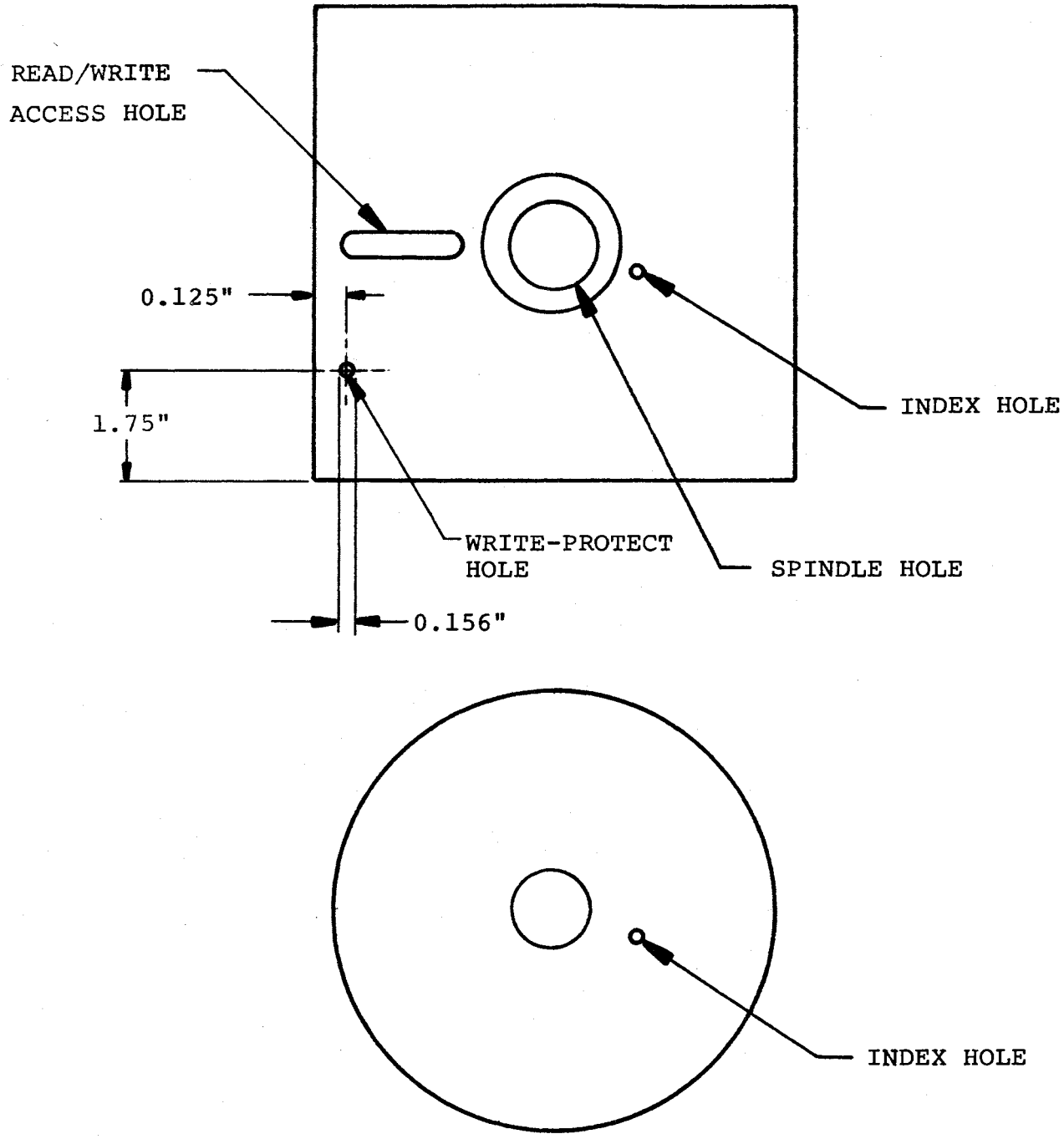


FIGURE 5-4
FLEXIBLE DISK CARTRIDGE

2. Write Fault (CDC only)

A logic 1 level on this line indicates one or more the following fault conditions:

-Write enable without head load.
-Write enable without write data.
Note: Erase current signal is derived internally on PCB from the write enable input line. Erase current is on automatically during writing.

A fault on this line can be cleared by a logic 1 on the Write Fault Reset line to the FDD.

3. Track 00

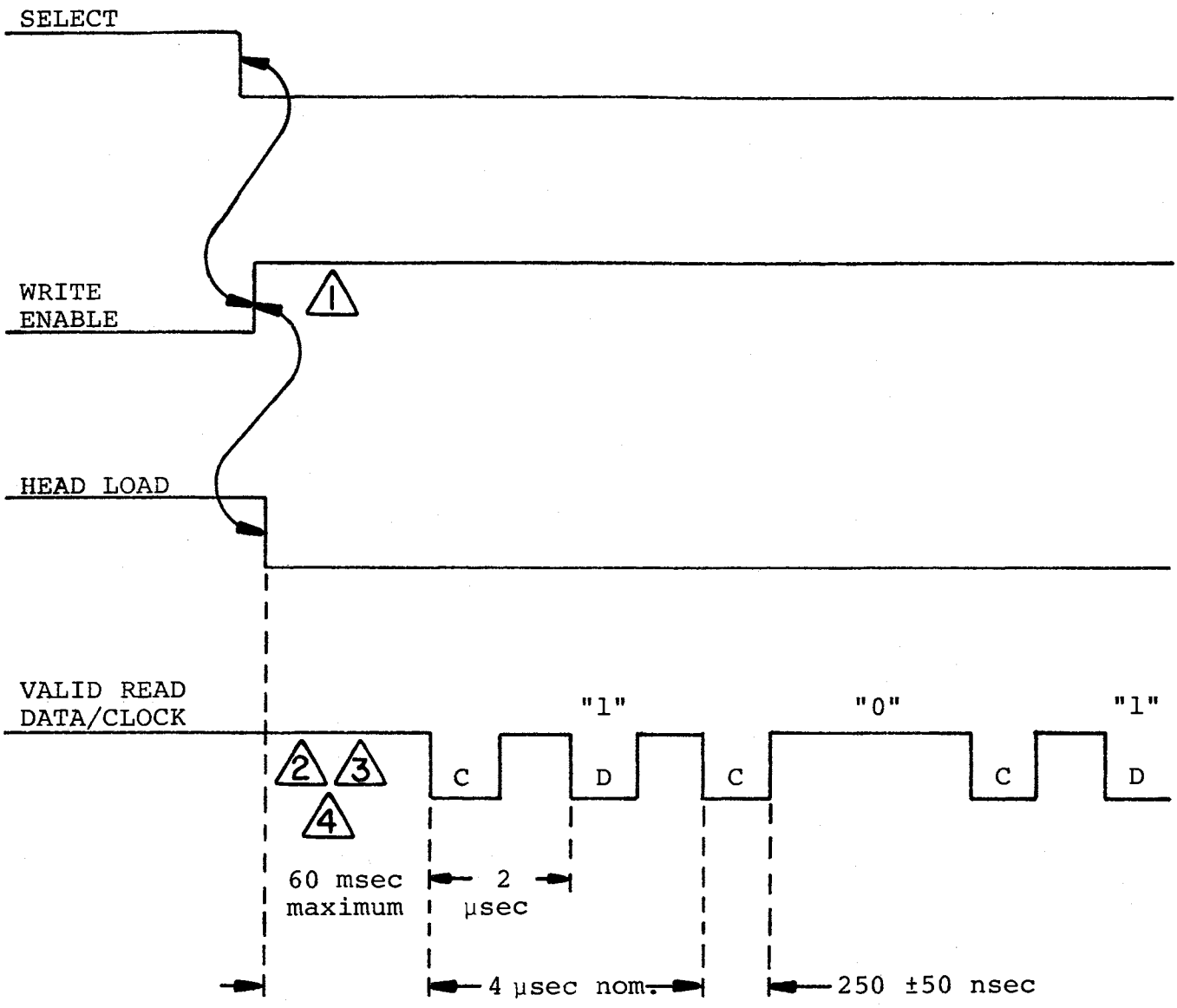
A logic 1 level on this line indicates that the head is positioned over track 00.

4. Separated Read Data (Single Density Only, refer to Figure 5.5).

This line contains the separated data information. A logic 1 level pulse of 250 nsec $\pm$ 20% corresponds to a data 1 bit read from the disk.

5. Clock (Single Density Only, refer to Figure 5-6)

This line contains the separated clock information. A logic 1 level pulse of 250 nsec $\pm$ 20% corresponds to a clock bit read from the disk.



1 When WRITE ENABLE IS HIGH read operation is implied.

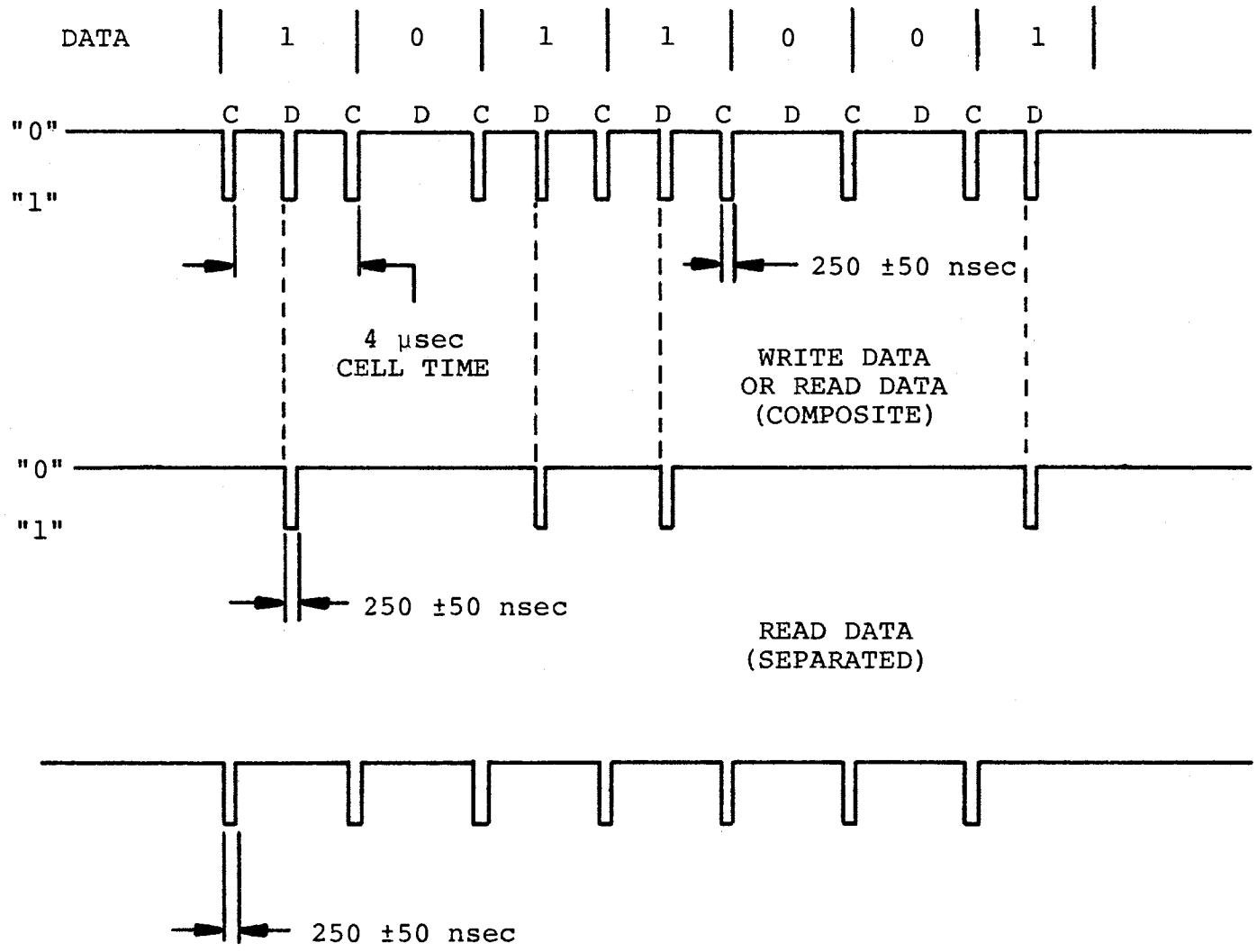
2 During this time interval READ DATA/CLOCK is to be ignored.

3 Subject to ±3.5% speed tolerance.

4 For product acceptance purposes any two adjacent bits may be subject to ±500 nsec bit shift from this nominal position.

FIGURE 5-5

READ DATA TIMING



CLOCK

NOTE: Synchronization shown between write data and read data/clock for illustration purposes only.

FIGURE 5-6

SINGLE DENSITY DATA TIMING

5.4 DISK CARTRIDGE STORAGE AND HANDLING

The disk performs well when given reasonable care. The same handling specified for computer magnetic tape should be followed. Some specific areas are as follows:

When not in the unit, keep the disk in the protective envelope. Place the disk in the envelope before writing on the label so that the label is visible through the cut-away front of the envelope.

Always handle the disk by the label area to avoid touching the mylar surface.

Keep all magnets away from disk. Magnetic fields can destroy recorded data on the disk.

Do not touch or attempt to clean the disk surface. Abrasion may result in loss of stored data.

5.5 ADDITIONAL INFORMATION

Before a disk drive can be used with the SBC Diskette Controller, the drive must be configured properly. There are three main areas to consider:

(1) Head Load Option
(2) Drive Select/Ready
(3) Daisy Chain Termination

The following sections describe in detail how to configure a CDC 9404 diskette drive or a Shugart 800/800R so that they will be compatible with the SBC Diskette Controller.

NOTE: If you have purchased an SBC 211, SBC 212, SBC 211/220V, or SBC 212/220V the drives were appropriately configured at Intel's factory before shipping. The following information then, is for your reference only. If you have purchased an SBC Diskette Controller (SBC 201) the following information is important for proper operation.

5.5.1 CDC DRIVES

5.5.1.1 HEAD LOAD

There are two metallic cups in the CDC Printed Circuit Board just to the left of the 50-pin header. These two points must be shorted together for the drive to operate properly. The cups are spaced so that they will accept a Cambion shorting plug #461-2871-01-03-16.

5.5.1.2 DRIVE SELECT/READY

There is a 10 position dual-in line switch on the CDC Printed Circuit Board to the left of the 50-pin header. The settings of the rockers on this switch determine the drive address and gate the ready signal to the appropriate pin. The drive will operate as "Drive Ø" if rockers 1 and 2 are on (all others off); the drive will operate as "Drive 1" if rockers 3 and 4 are on (all others off). Any other settings will cause improper operation.

5.5.1.3 "DAISY CHAIN" TERMINATION

The CDC drives are meant to be used in a daisy chain configuration. The lines are terminated with a resistor pack which mounts in the 16-pin socket located to the left of the 50-pin header on the drive Printed Circuit Board. One and only one drive must have the terminating resistor pack installed. Common practice is to terminate the drive which is addressed as "Drive Ø".

5.5.2 SHUGART DRIVES

5.5.2.1 HEAD LOAD

The Shugart drive as shipped from the factory is configured for the appropriate head load option.

5.5.2.2 DRIVE SELECT/READY

Drive selection is made by slip-on shorting plugs on the drive Printed Circuit Board. To operate as "Drive Ø" with the SBC Diskette Controller, jumper "DS1" on the drive PCB should be shorted. To operate as "Drive 1" with the SBC Diskette Controller, jumper "DS2" on the drive PCB should be shorted. One and only one DS jumper should be shorted.

Radial ready select is made by cutting and jumpering traces on the drive Printed Circuit Board. If the drive is addressed as "Drive Ø", cut the etch between the pads labeled "R" and between the pads labeled "RR" on the PCB; add a jumper from the board side of R to output pad 12. If the drive is addressed as "Drive 1", cut the etch between the pads labeled "R" and the pads labeled "RR" on the PCB; add a jumper from the board side of "R" to output pad 14.

5.5.2.3 "DAISY CHAIN" TERMINATION

The four daisy chained lines are terminated by slip-on plugs at T3, T4, T5, and T6. One and only one drive must have these terminating plugs installed. Common practice is to terminate the drive which is addressed as "Drive Ø".

CHAPTER 6
DISKETTE SYSTEM MICROPROGRAM

6.1 INTRODUCTION

The Diskette Controller μcode consists of 512 words x 32 bits. The bit description can be found in Table 3-2. The μcode is organized functionally as modules. There are two types of modules - primary and subroutine. Primary modules, in general, implement high level functions while subroutines implement specific lower level functions. A specific subroutine may be invoked by many primary modules, but no subroutine can itself invoke another subroutine.

The primary modules within the μcode are as follows:

INIT	- Initialization
ML	- Mainline
MAL	- Load M.A. Lower
STIO	- Load M.A. Upper and Start I/O
RR	- Read Result Byte
IOPB	- IOPB Loader/Op Decode
FIN	- I/O Finish
SEEK	- Seek
FMT	- Format
REC	- Recalibrate
VERF/RD	- Verify/Read
WDEL/WT	- Write Deleted/Write

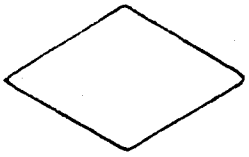
The subroutine modules within the μcode are as follows:

CHKR	- Address Paramter Checker
INC	- Read Next Memory Word
WDAT	- Write Data Field
WCURR	- Write Current Check
TO	- Time Out
AM	- Address Mark Detect

MVR - Head Stepper
TNIBS - Read Disk Byte
ID - Process Address (ID) Field
WADD - Write Address Field

The following conventions will be used throughout:


- Denotes a command or data transfer.


- Denotes a decision block.


- Denotes a transfer of control to another module.

6.2 MICROPROGRAM MODULE DESCRIPTION

The following flow charts describe each module in detail.

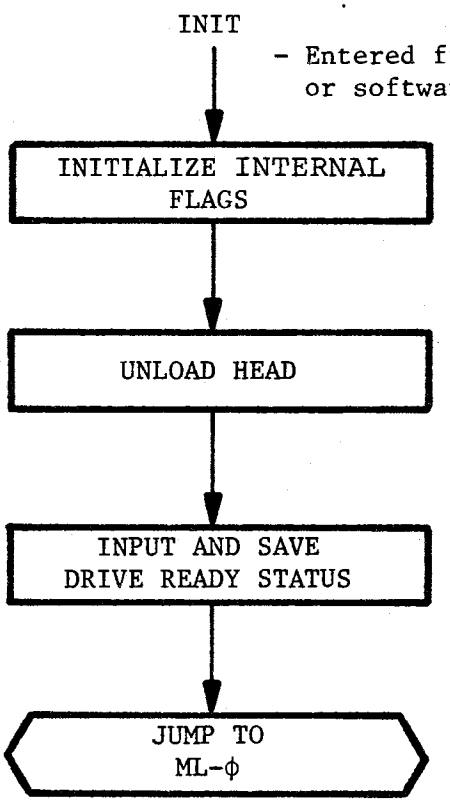


FIGURE 6-1

INITIALIZATION

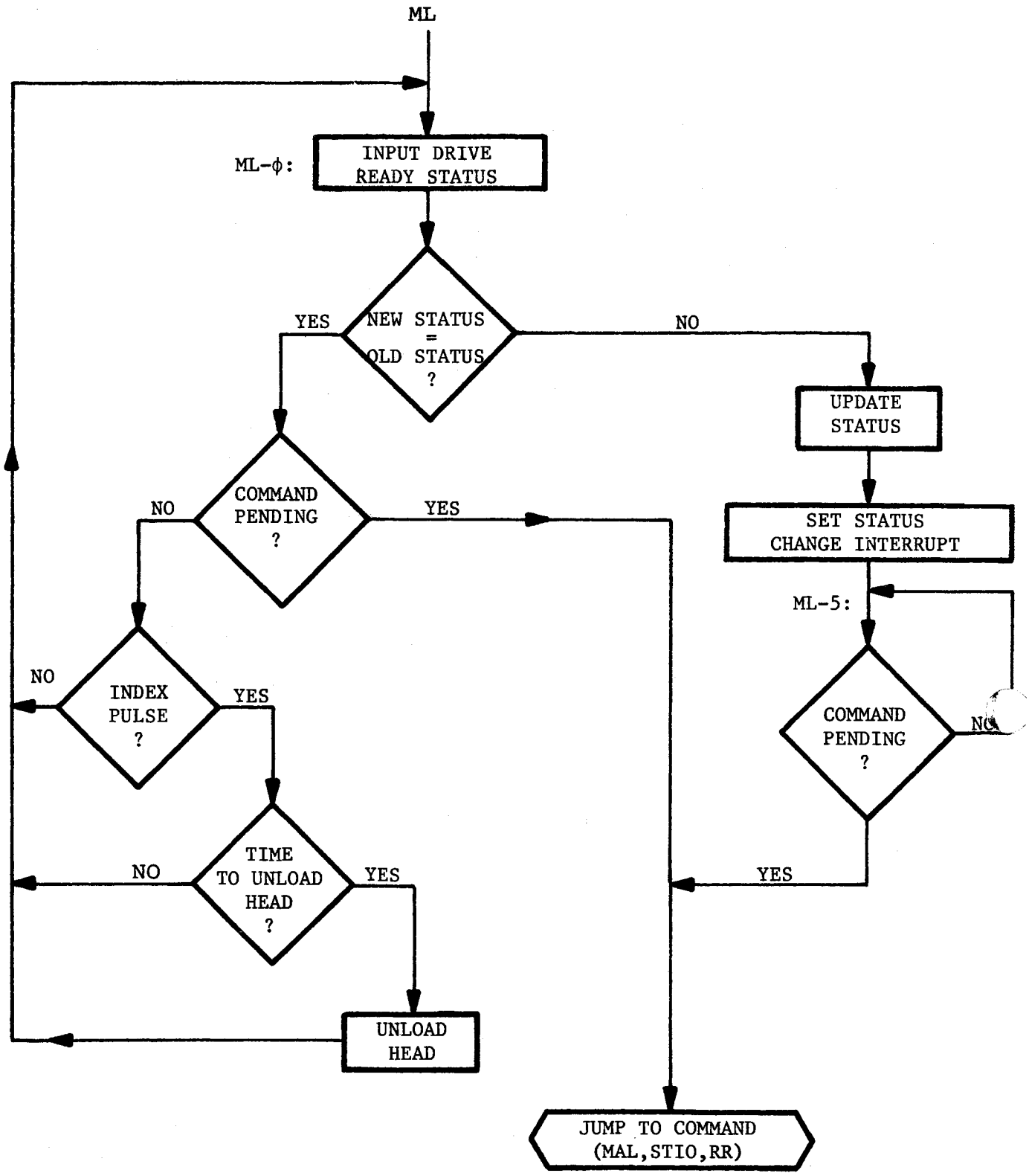


FIGURE 6-2

MAINLINE

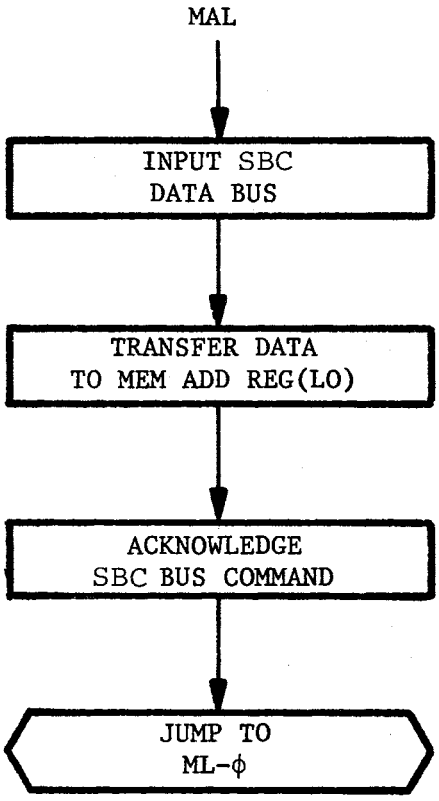


FIGURE 6-3

LOAD MA LOWER

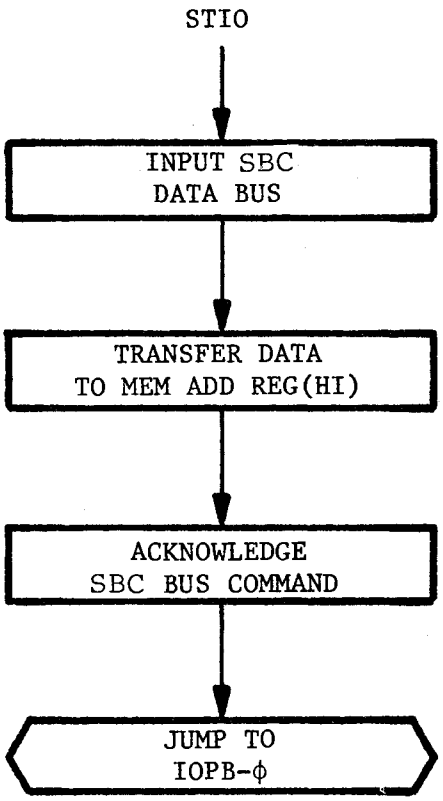


FIGURE 6-4
LOAD MA UPPER AND START I/O

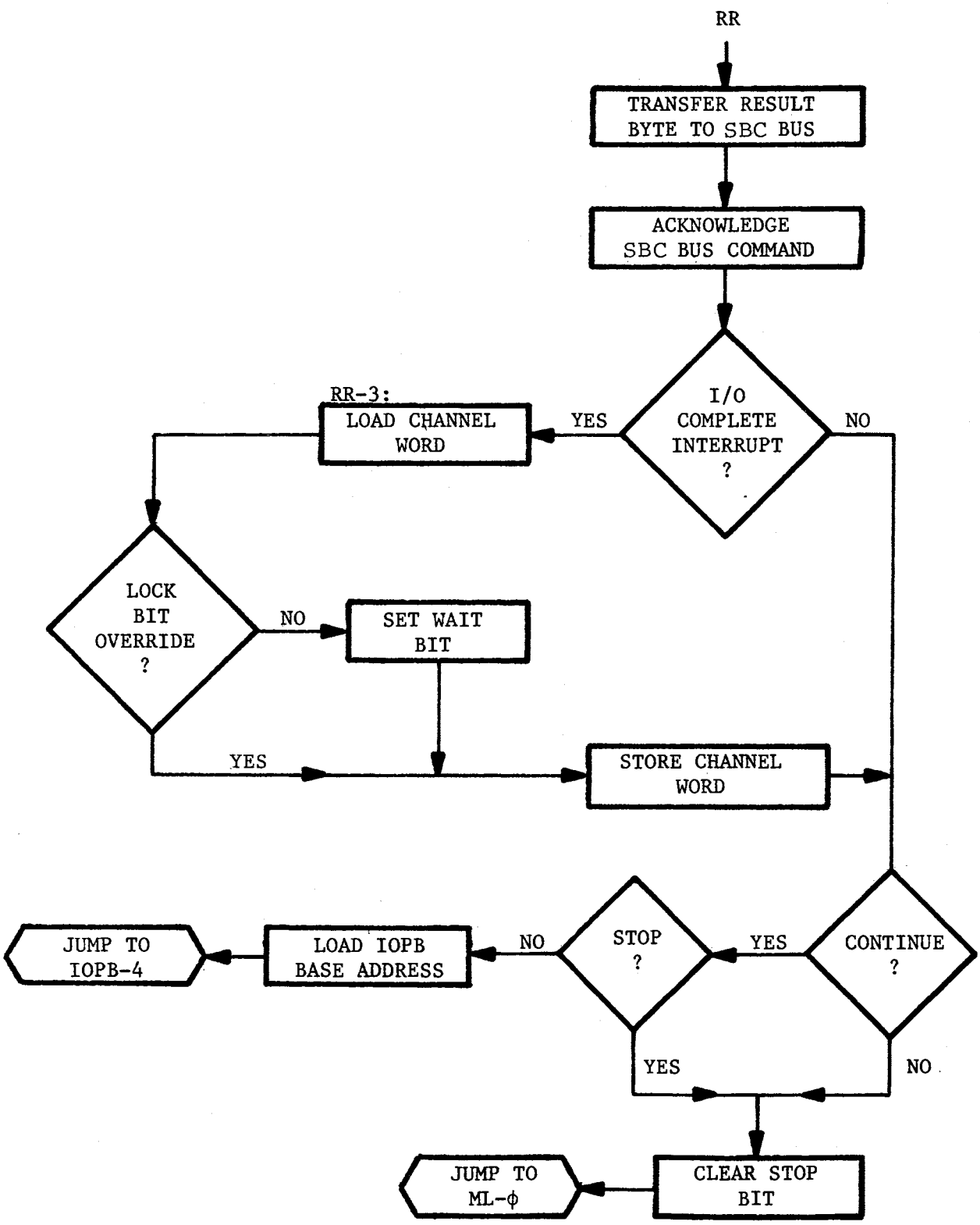


FIGURE 6-5

READ RESULT BYTE

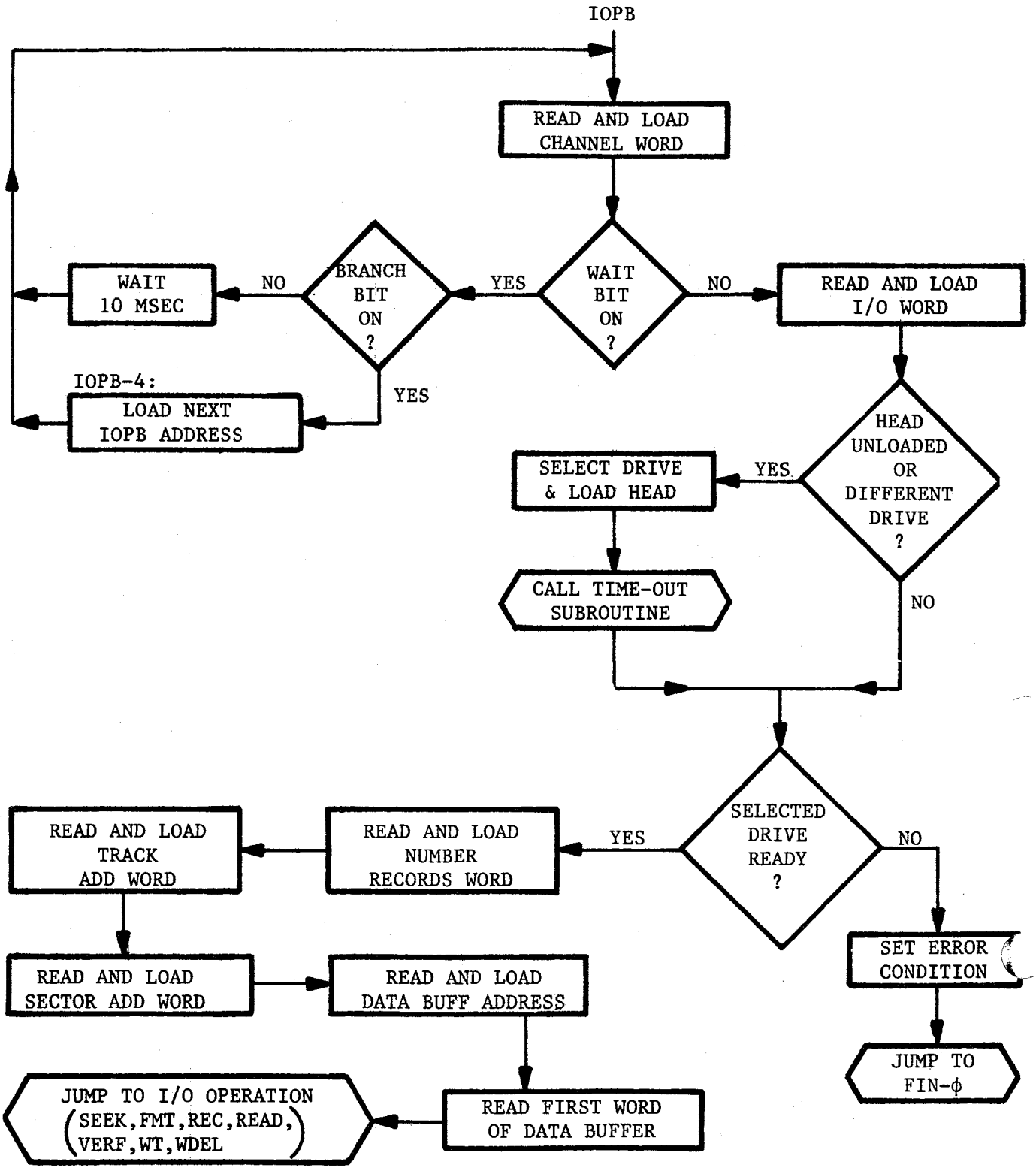


FIGURE 6-6

IOPB LOADER/OP DECODE

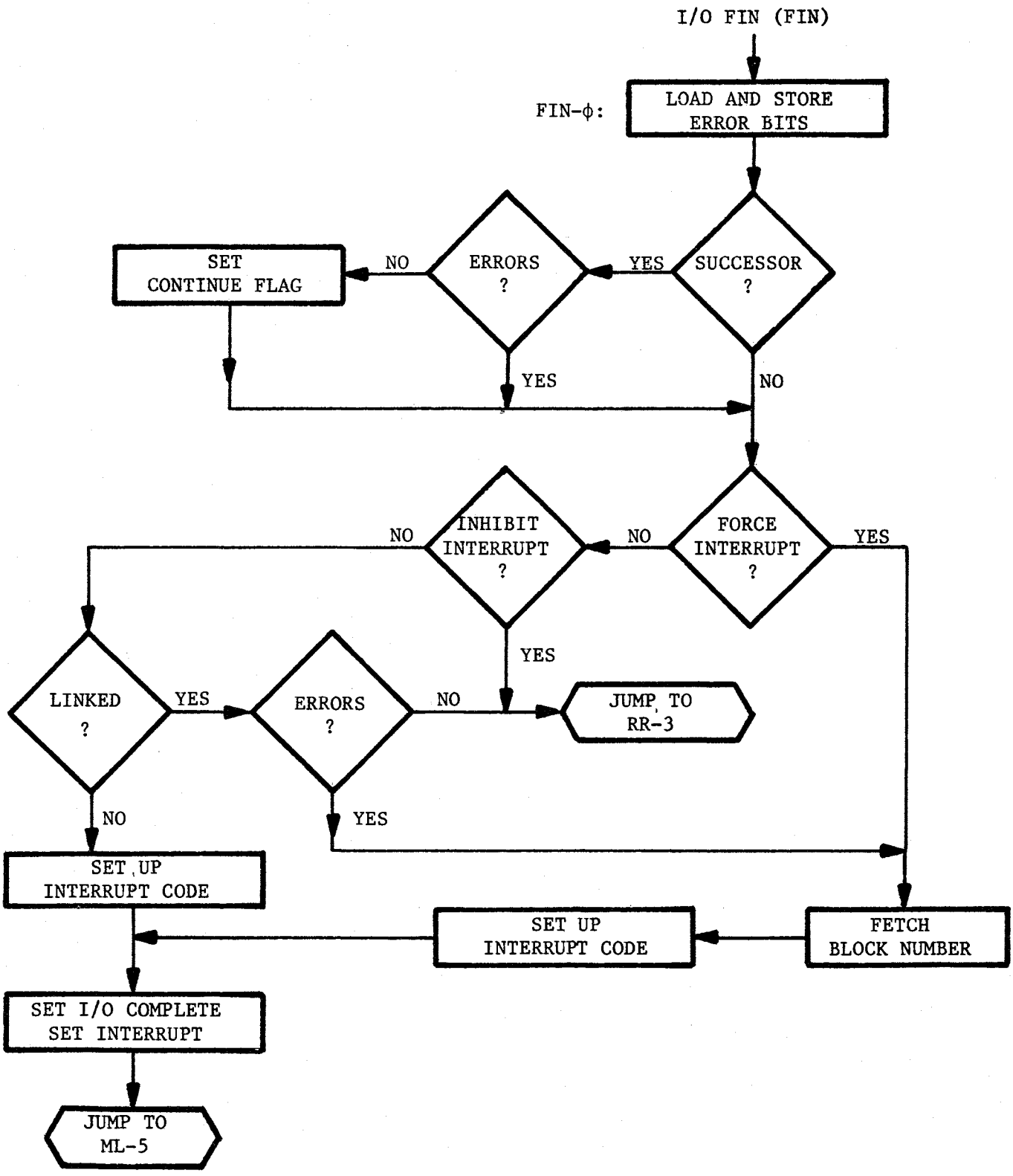


FIGURE 6-7

I/O FINISH

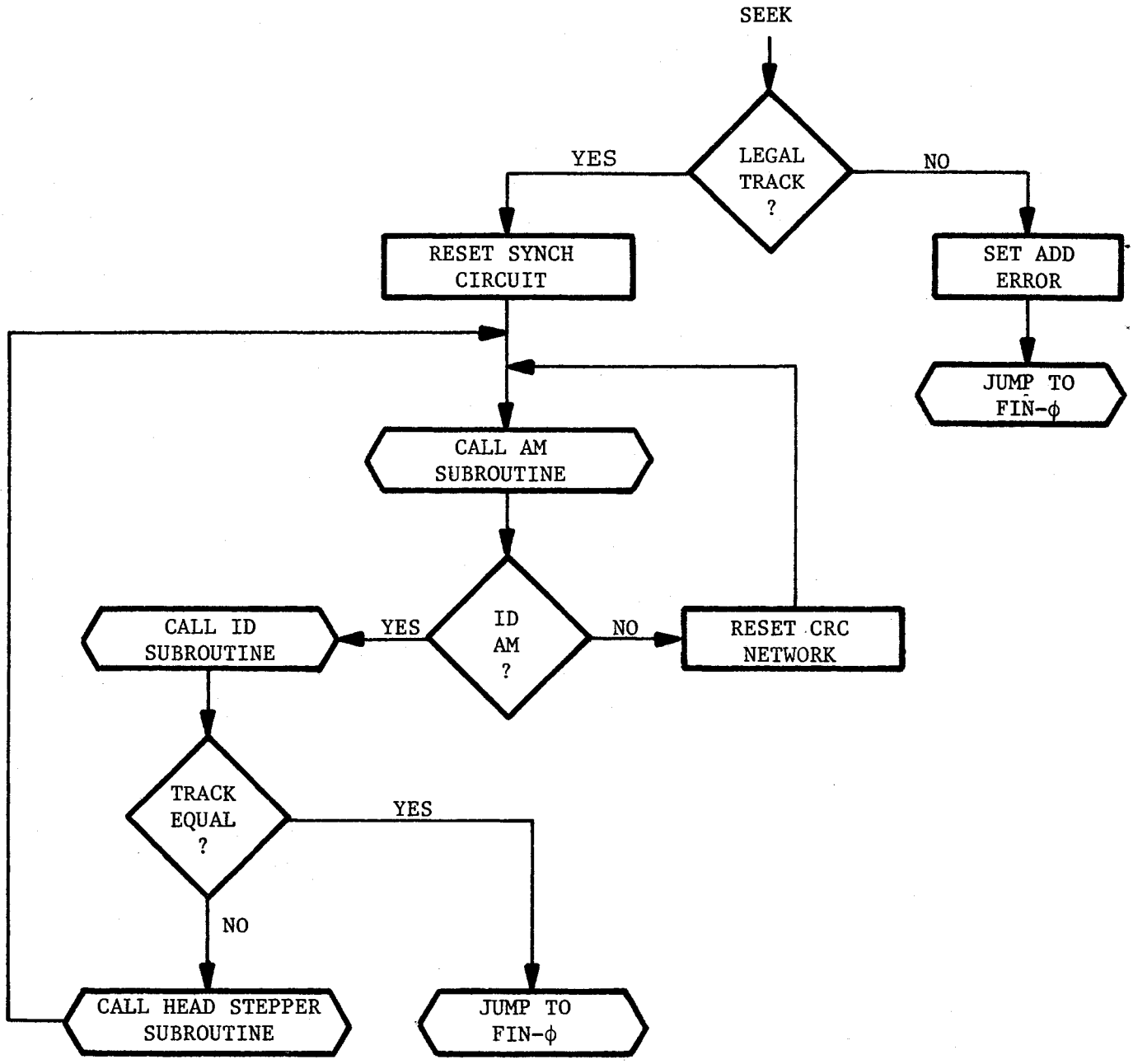


FIGURE 6-8

SEEK

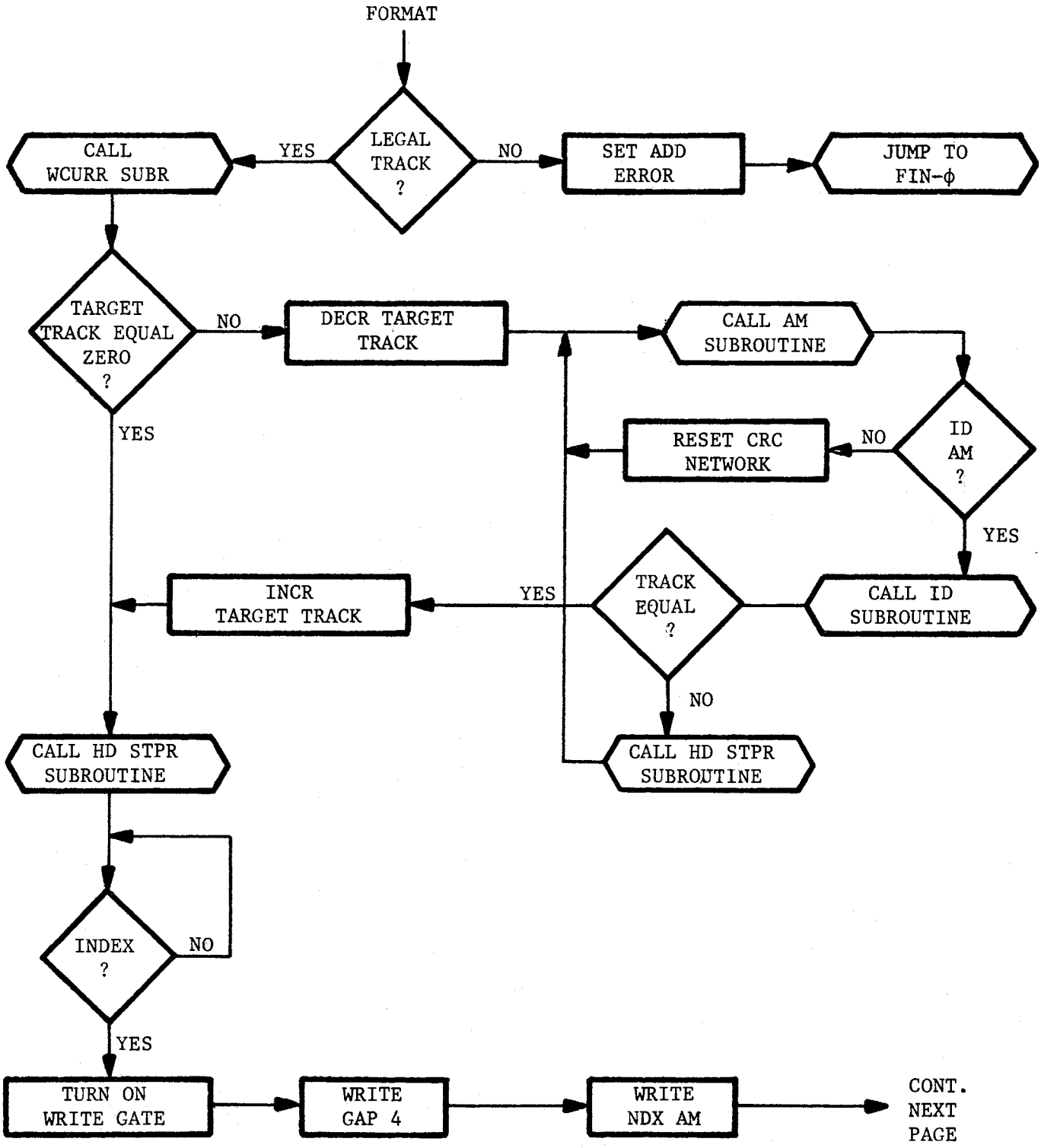


FIGURE 6-9

FORMAT

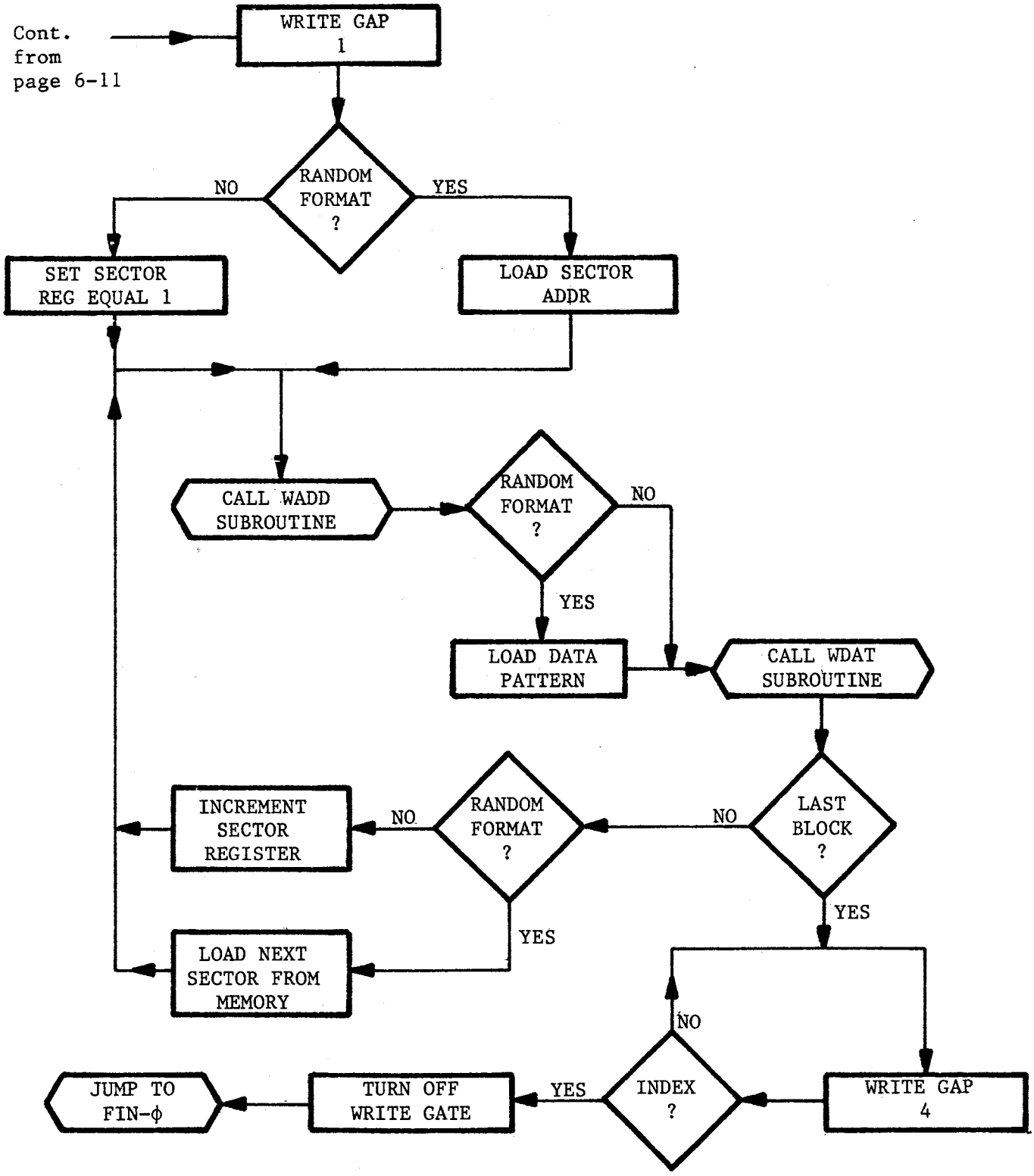


FIGURE 6-9 (CONTINUED)
FORMAT

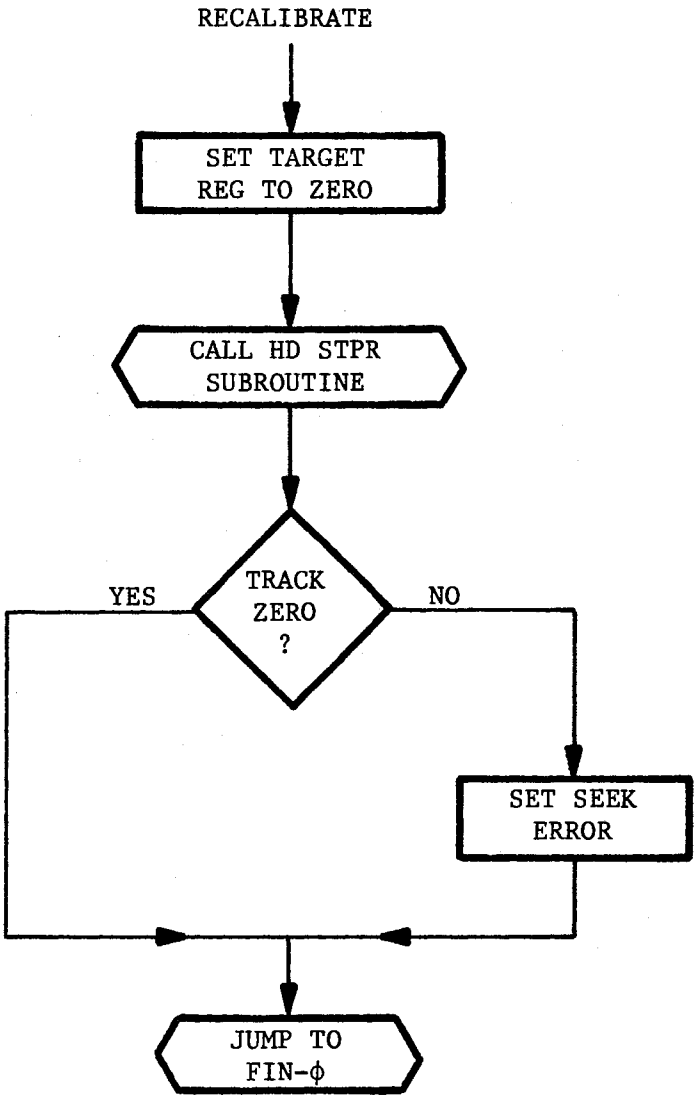


FIGURE 6-10
RECALIBRATE

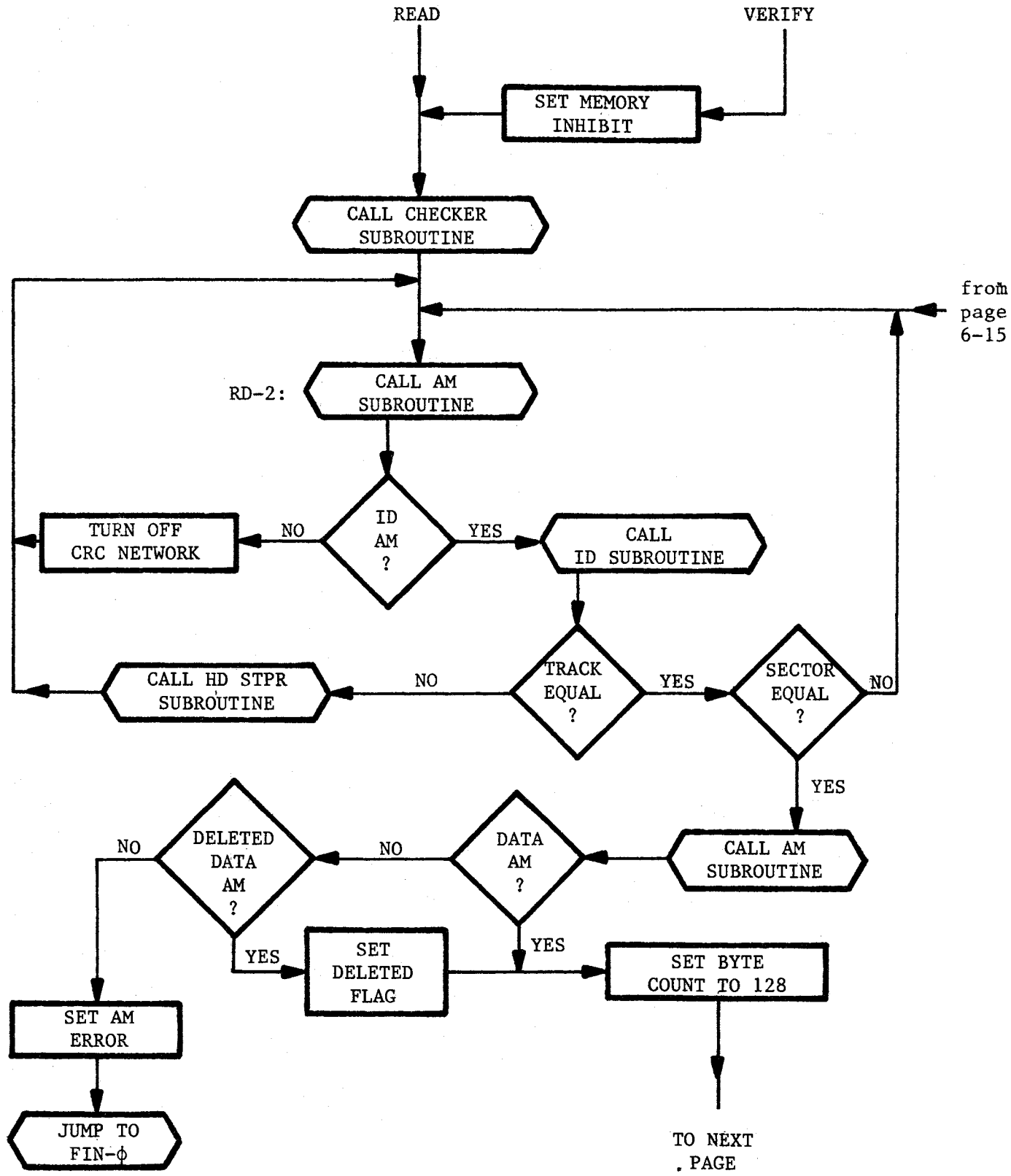


FIGURE 6-11
VERIFY/READ

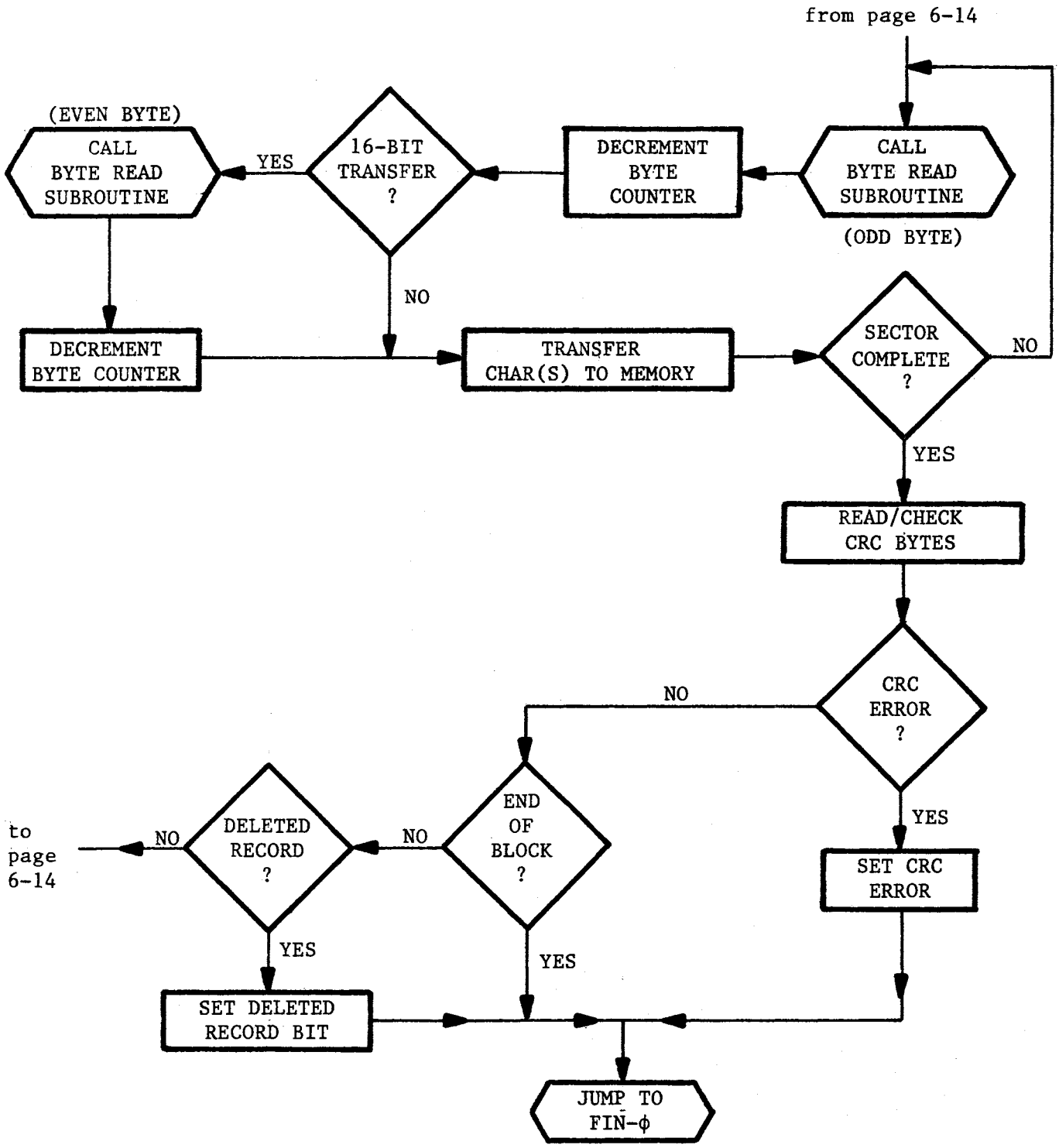


FIGURE 6-11 (CONTINUED)
VERIFY/READ

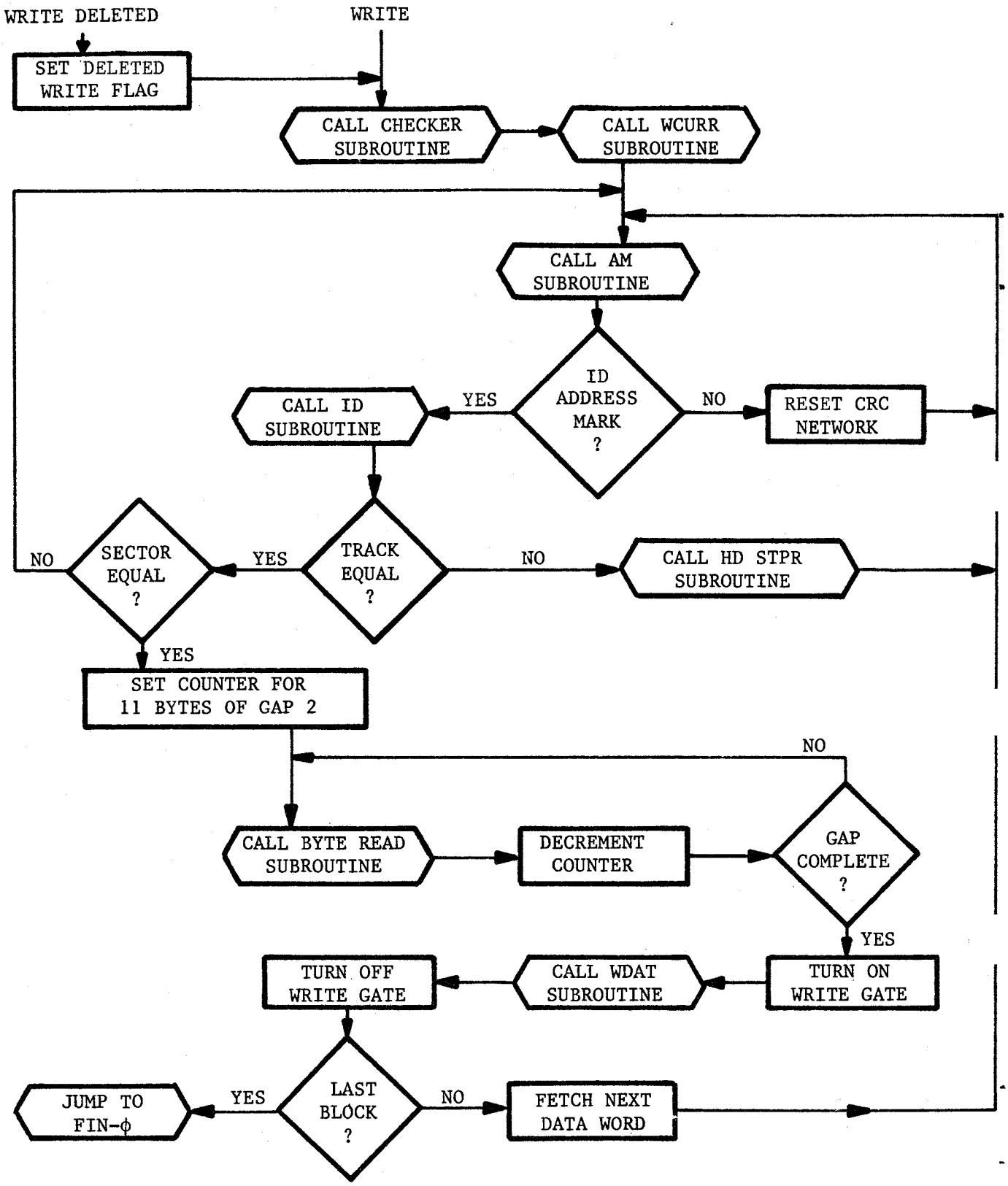


FIGURE 6-12

WRITE DELETED/WRITE

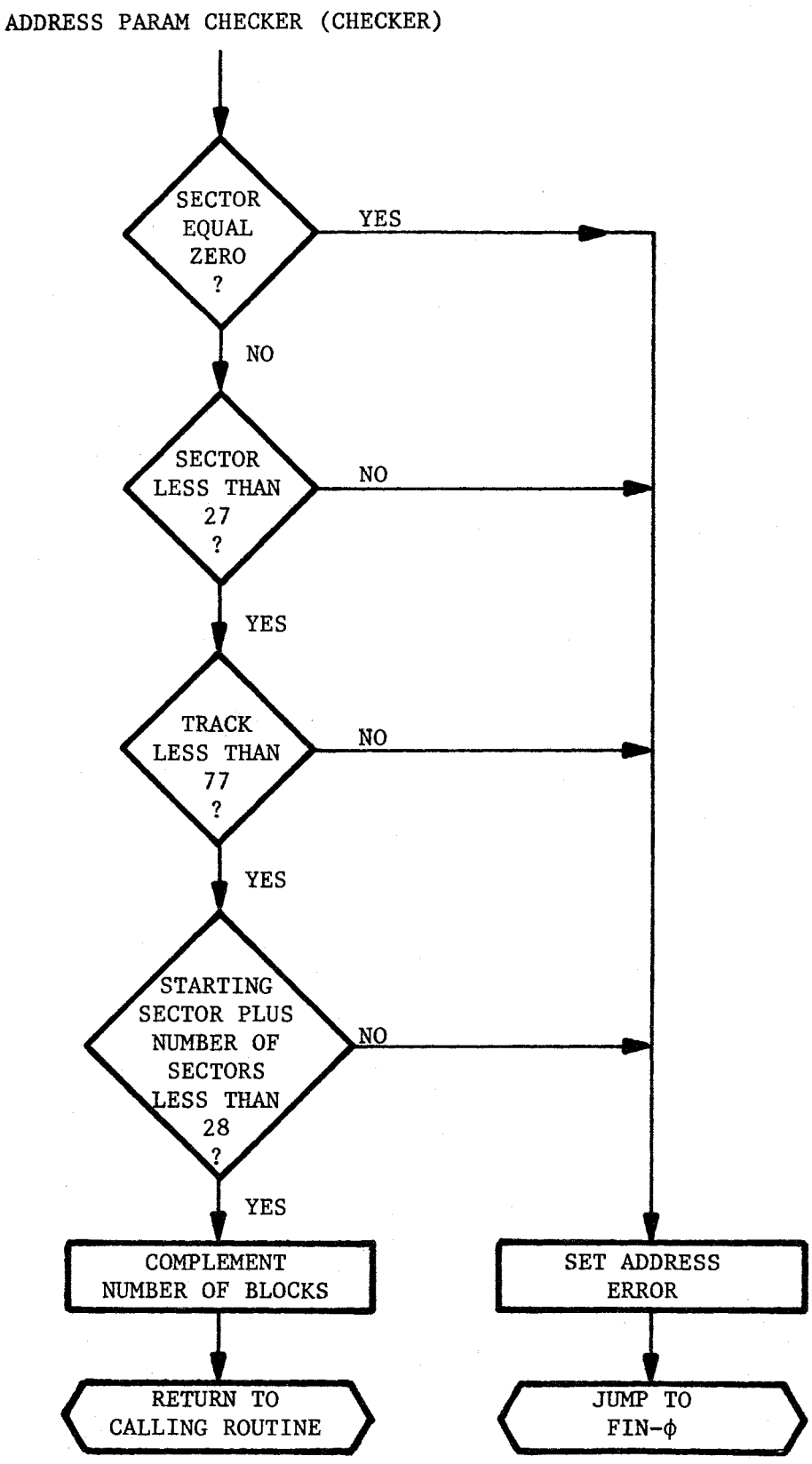


FIGURE 6-13

ADDRESS PARAMETER CHECKER

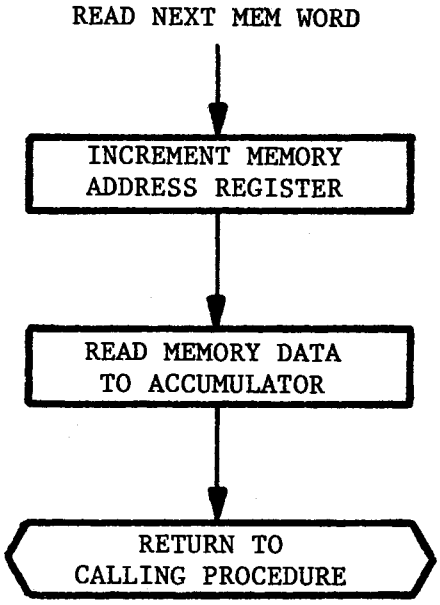


FIGURE 6-14
READ NEXT MEMORY WORD

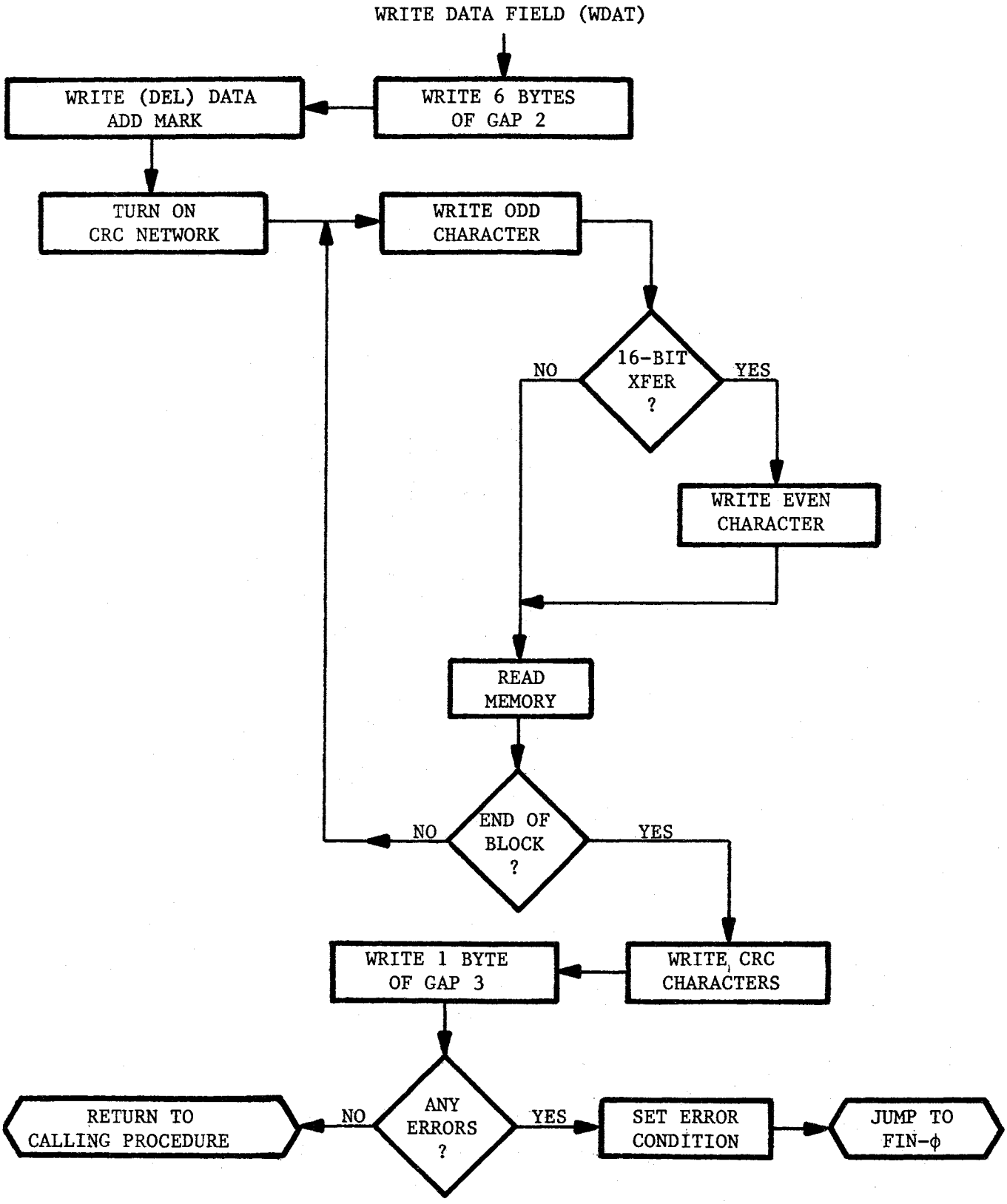


FIGURE 6-15

WRITE DATA FIELD

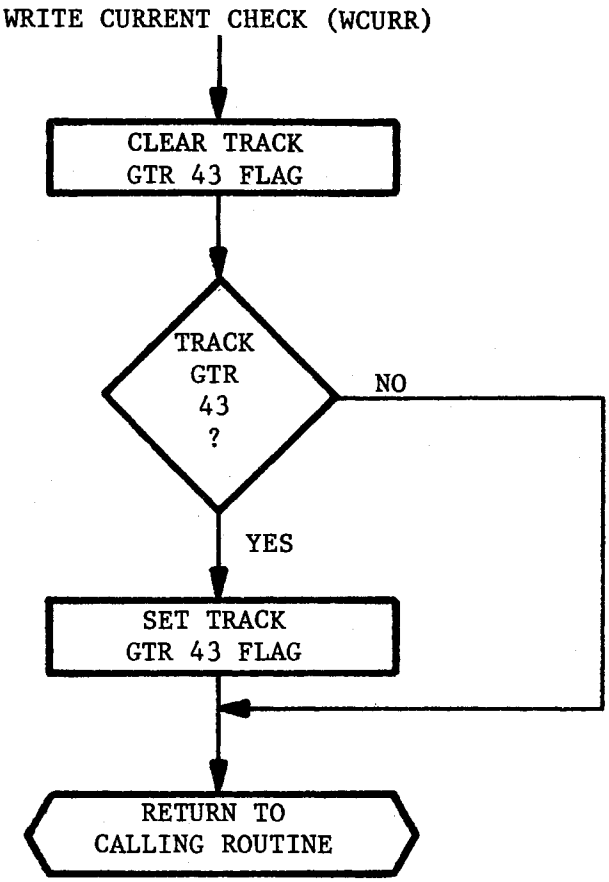


FIGURE 6-16
WRITE CURRENT CHECK

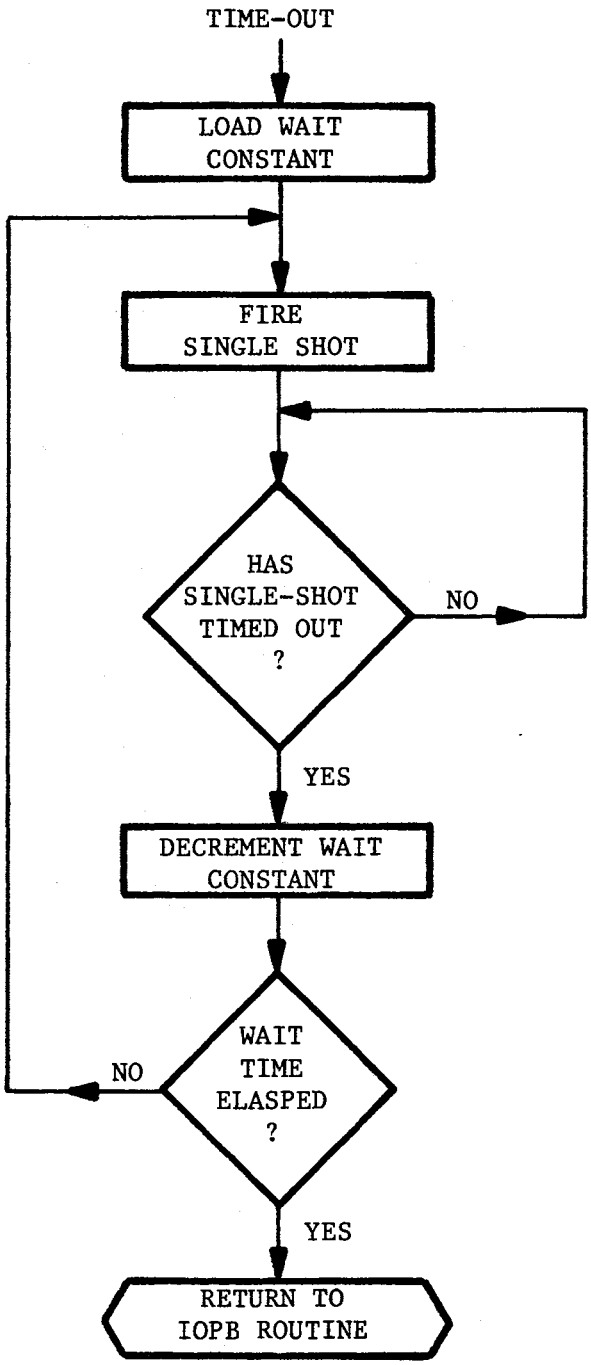


FIGURE 6-17

TIME-OUT

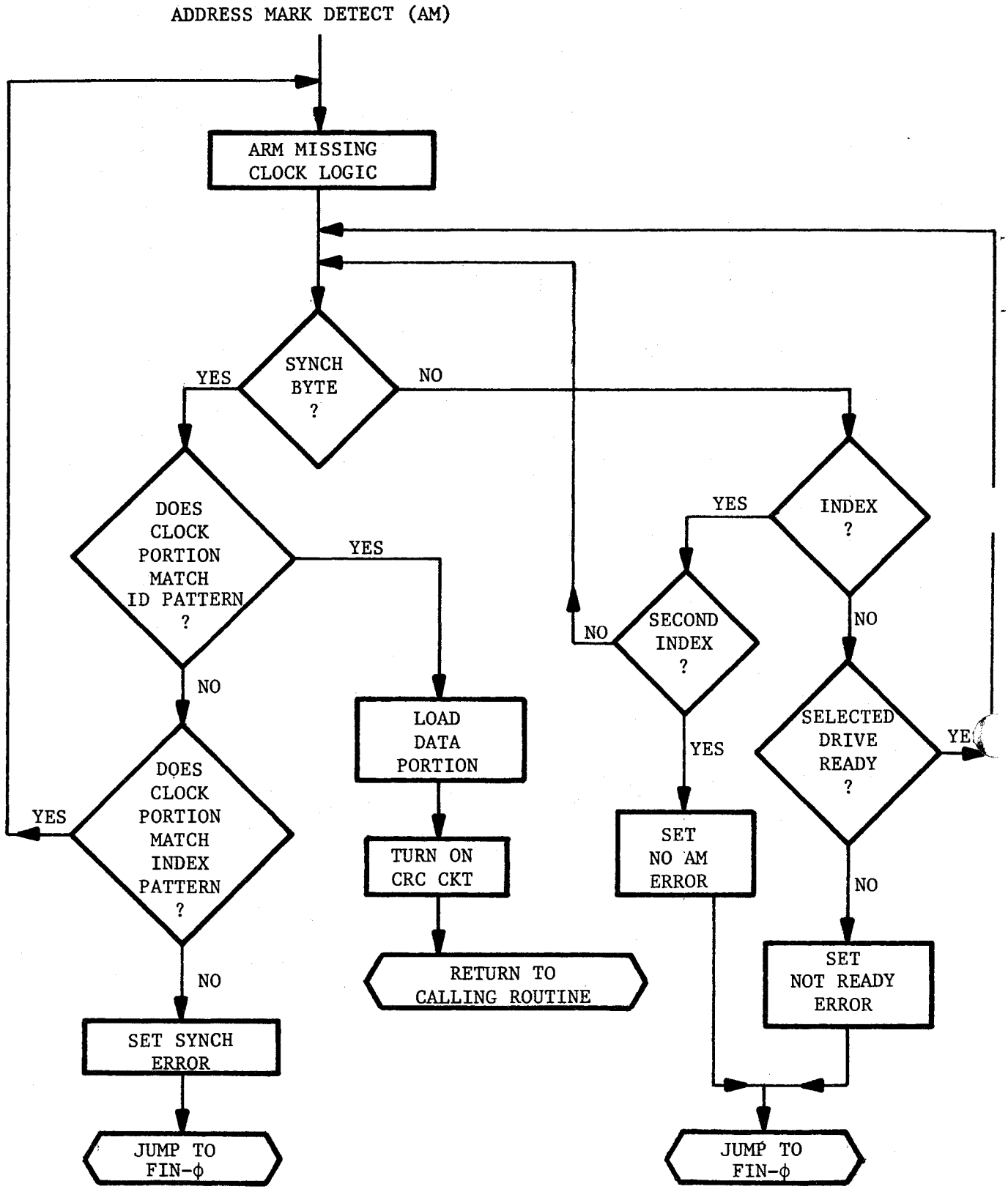


FIGURE 6-18

ADDRESS MARK DETECT

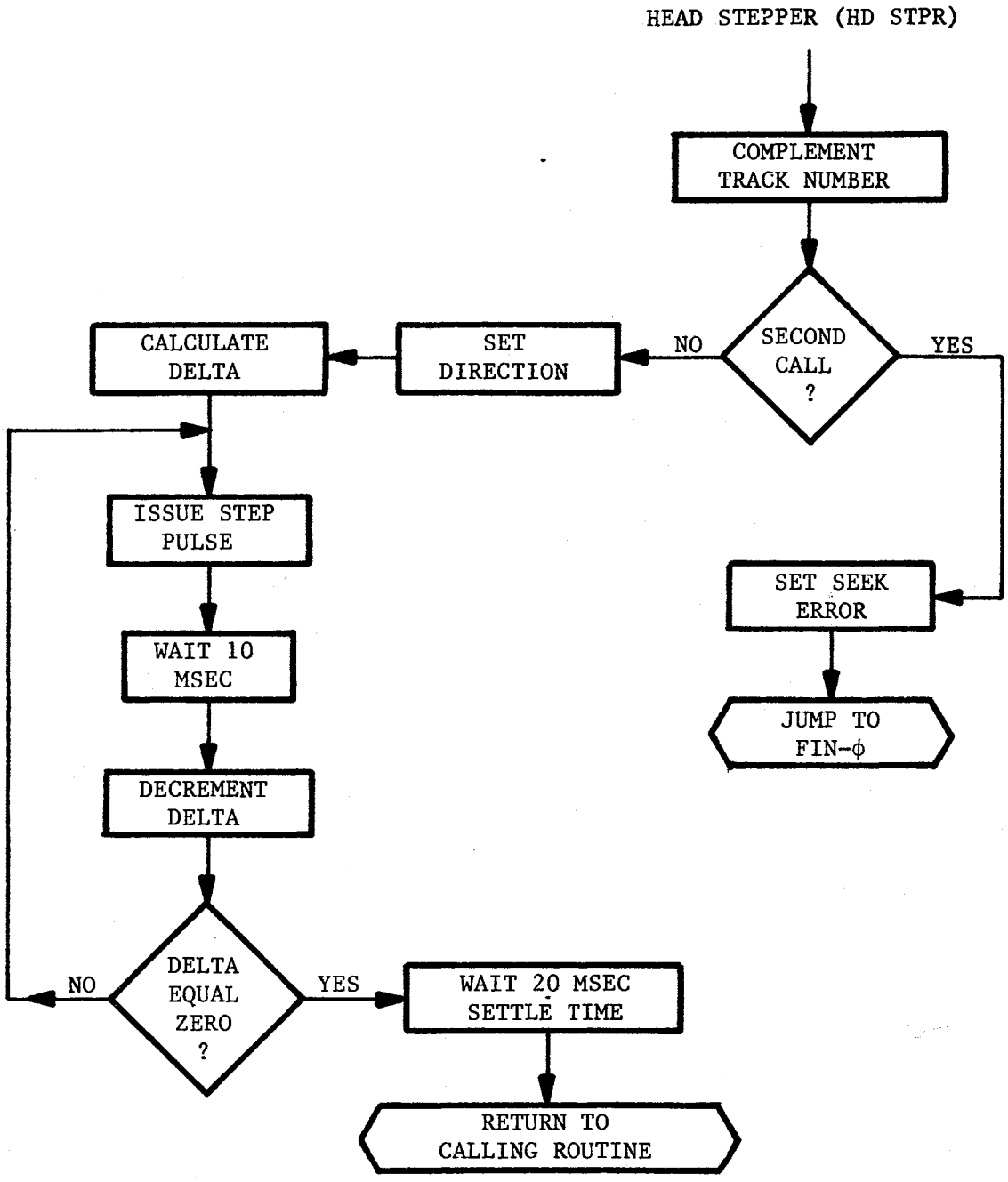


FIGURE 6-19
HEAD STEPPER

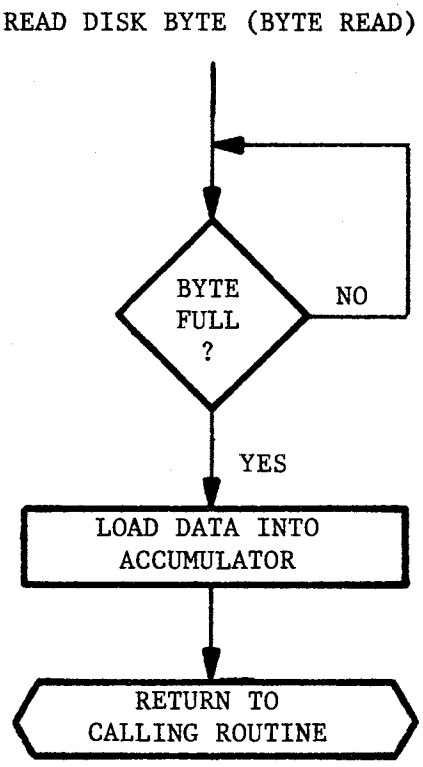


FIGURE 6-20
READ DISK BYTE

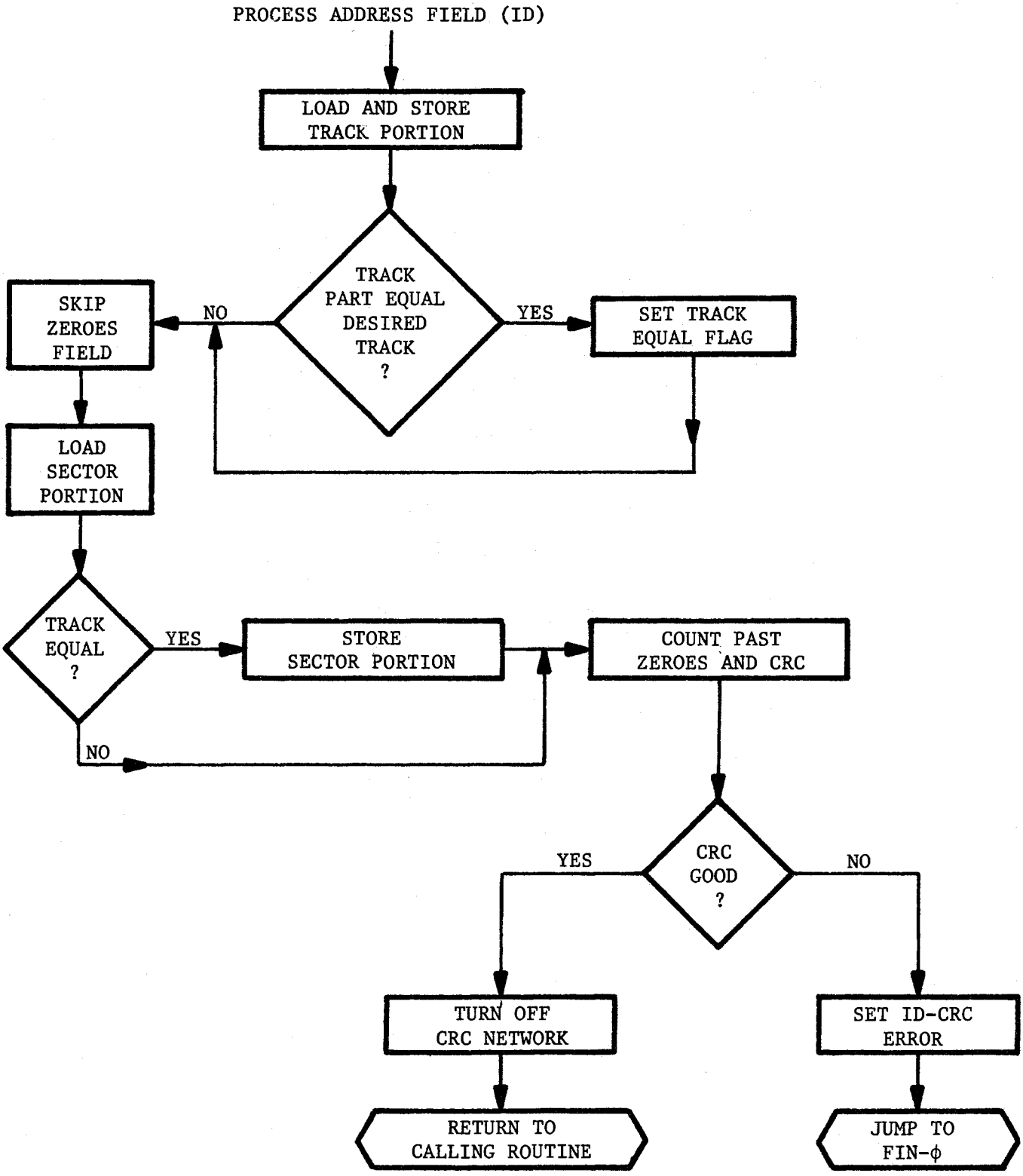


FIGURE 6-21
PROCESS ADDRESS FIELD

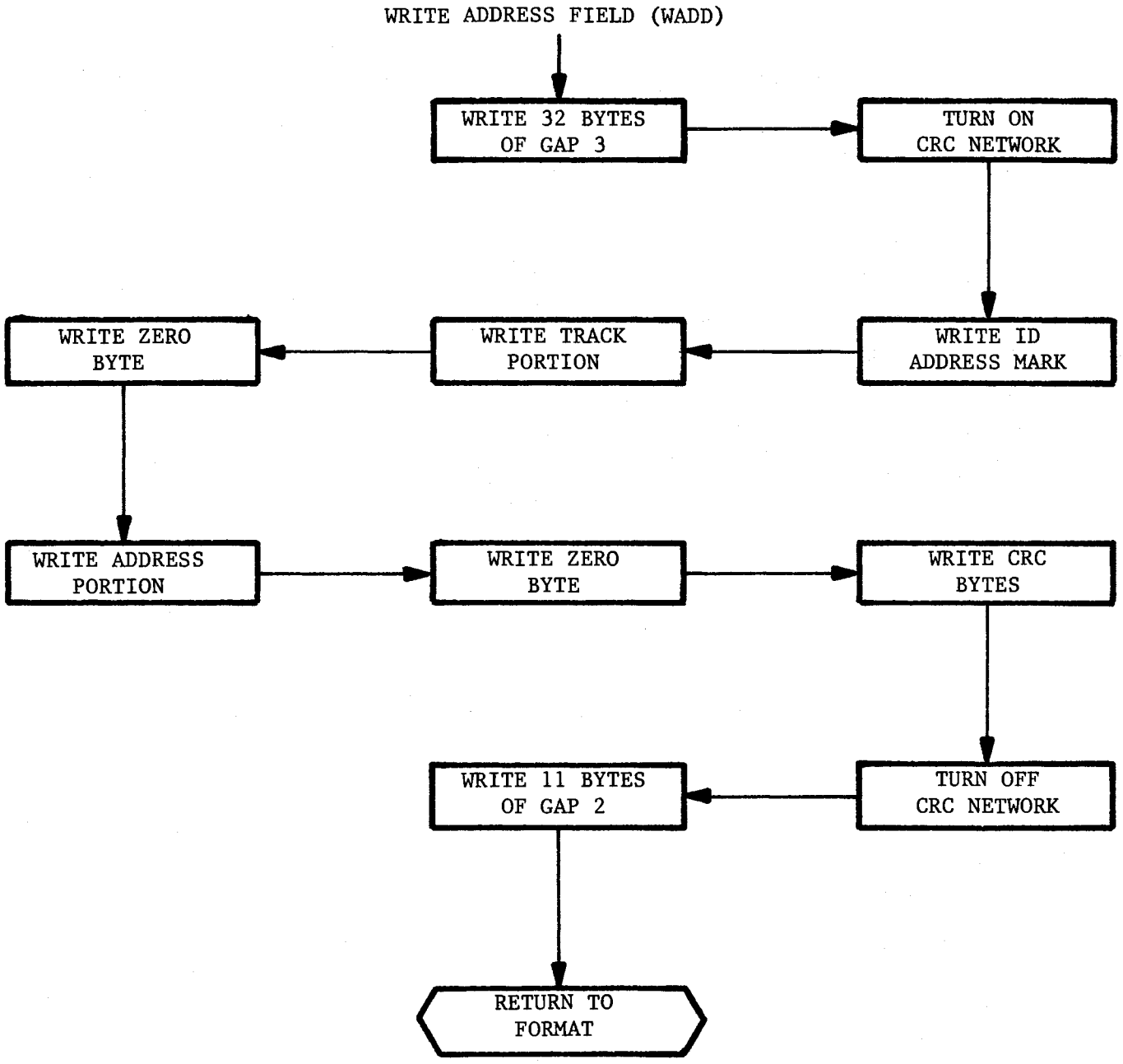


FIGURE 6-22

WRITE ADDRESS FIELD

CHAPTER 7
UTILIZATION

This chapter provides the necessary information to install and fully utilize the capabilities of the Diskette System. In installing the Diskette System, the user should consider:

1. Environmental extremes
2. Mounting recommendations
3. Electrical connections
4. Base address selection
5. Interrupt level selection

The following sections of this chapter will deal with these considerations.

7.1 ENVIRONMENTAL EXTREMES

Temperature extremes can cause instability, or result in permanent damage to the circuits or the drive mechanisms. Ambient temperature must therefore be maintained within the limits of 50^{o} to 100^{o}F (10^{o} - 38^{o}C), 20% to 80% relative humidity with maximum wet bulb temperature of 80^{o}F.

Exercise caution in locating the system, giving particular attention to radiant and conductive sources of heat. Remember that the system itself will contribute some heat to the environment (314 BTU/hour for each diskette drive). Provide adequate ventilation to permit the convective dissipation of heat from the system components.

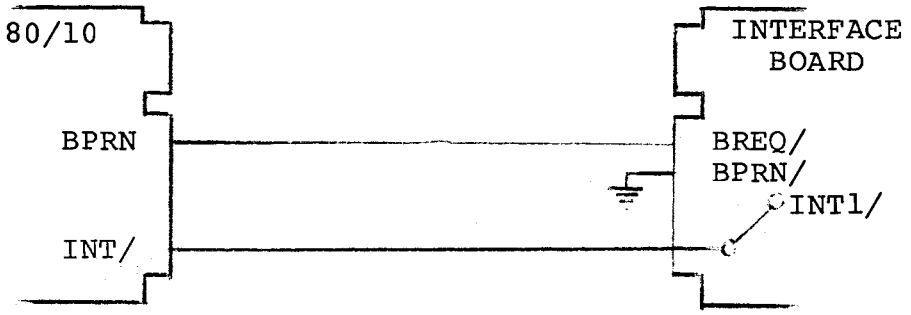
7.2 MOUNTING RECOMMENDATIONS

Controller: The Channel Board and Interface Board (jointly referred to as the Diskette Controller) are to be inserted into a backplane. Both boards are designed to plug directly into three standard PC-edge connectors. An 86-pin connector and a 60-pin connector are located on the bottom edge of the board and plug into the motherboard assembly. A 100-pin connector is on the opposite edge of the boards. The connectors serve as a mounting, as well as an electrical junction (see Section 7.3). Additional protection is provided by the guide slots in the System 80 or SBC 604/614 cardcage. The Channel and Interface Boards are each 12.00" x 6.75".

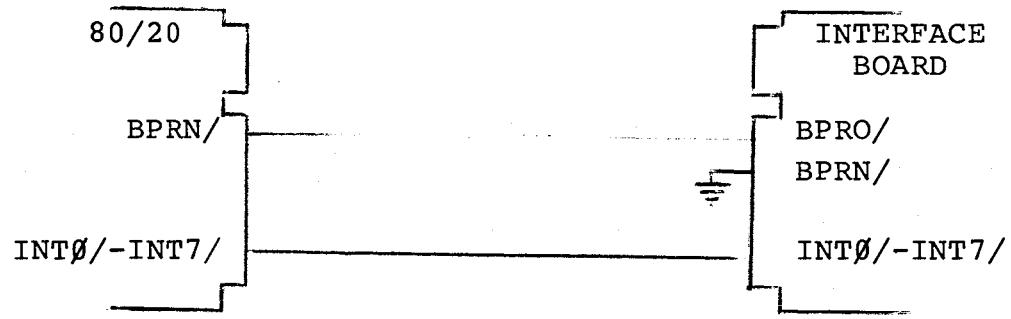
Before the controller boards are installed in the System, the 60-pin dual auxiliary connector assembly (4001066) must be installed. The controller boards then plug into the 2 card slots having the dual auxiliary connector.

To operate with either the SBC 80/10 or SBC 80/20, some simple jumpering is required.

SBC 80/10 Operation:



SBC 80/20 Operation:



7.3 ELECTRICAL CONNECTIONS

The Channel and Interface Boards each communicate with the Intel® system bus through a standard 86-pin, double-sided PC edge connector (P1), 0.156" contact centers (see Figure 7-1). CONTROL DATA CORPORATION's VPB01E43A00A1 is one suitable type of connector for P1. The two controller boards communicate with each other via a 60-pin, double-sided PC edge connector, 0.1" contact centers (see Figure 7-1). CDC97169001 is one suitable type of connector for P2. Both boards also include a 100-pin, double-sided PC edge connector (J1), 0.1" contact centers (see Figure 7-1). VIKING 3VH50/1JN5 is one suitable type of connector for J1. The Channel Board only uses its J1 connector as a means of accessing various test points on the board. The Interface Board, however, communicates with the diskette drive(s) via its J1 connector. The Interface Board's J1 connector also allows access to various test points. Pin allocations for each of the connectors on the Interface Board are provided in Section 4.3. The same information for the Channel Board is provided in Section 3.3.

The cables required for interfacing to the diskette drives are either Intel or customer supplied.

If you ordered an SBC 211 or SBC 212, you received 2 cables. The card edge connector end of the Floppy Disk Controller Cable plugs into the J1 connector on the Interface Board. The other end of the Controller Cable mates with the Floppy Disk Peripheral Cable. The other end of the Peripheral Cable plugs into the rear panel of the Diskette Drive enclosure.

If you ordered an SBC 201, you must build your own cables or order SBC 951 (Shugart compatible cables) or SBC 952 (CDC compatible cables) from Intel.

The SBC 951 and SBC 952 each provide two cables - a round cable and a flat cable. The cables mate to each other and the flat cable mates directly with the drives while the round cable mates to the Interface Board.

The Diskette enclosure also includes an AC power cord which must be plugged into a convenient AC source.

CAUTION: If you have ordered an SBC 211 or SBC 212 the Diskette enclosure is shipped from the factory as either a 110V/60Hz system or a 220V/50Hz system. Consult the shipping list and customer letter before plugging in the AC line cord. Using incorrect line voltage/frequency results in improper operation and potential damage to the diskette drives.

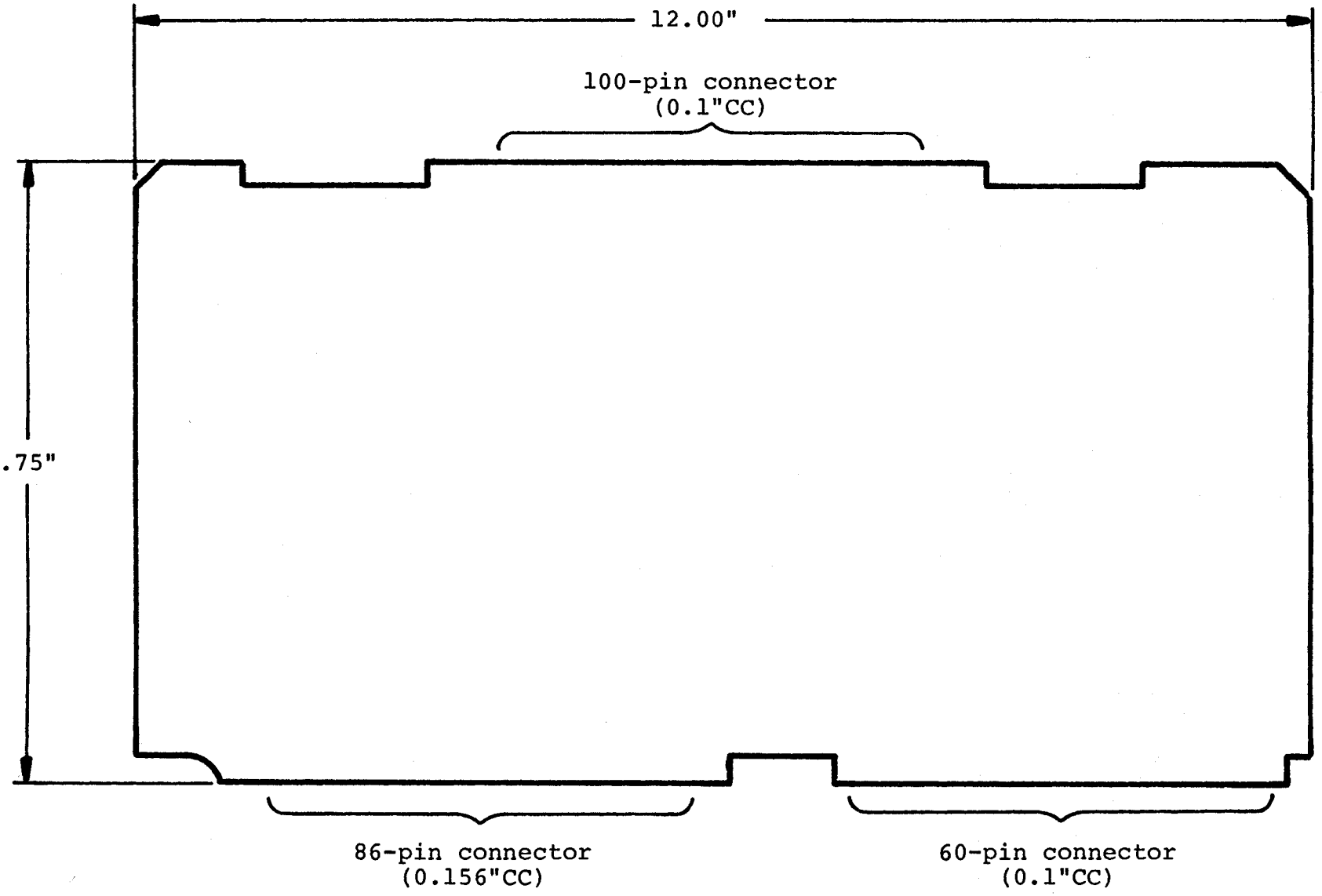


FIGURE 7-1

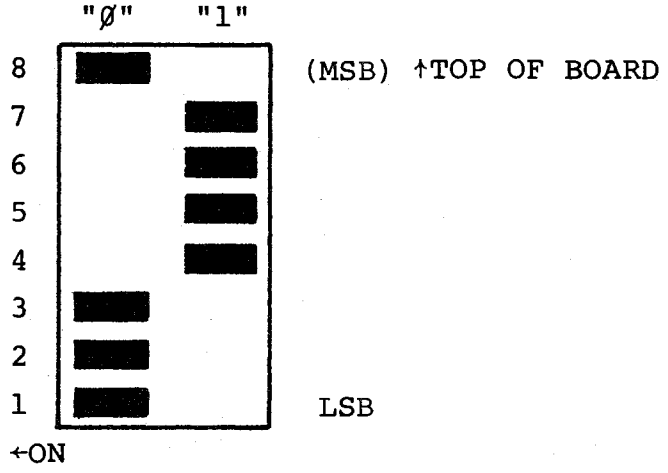
CONNECTORS ON THE CHANNEL AND INTERFACE BOARDS

7.4 BASE ADDRESS SELECTION

The user must assign a base address to the Diskette Channel. The base address is defined by the five most significant bits of the eight-bit I/O port address. The three least significant bits, then, can be used to differentiate between eight input or eight output channel commands. When the CPU accesses the Diskette Channel by executing an I/O instruction, the base address (BASE) is used to select the Diskette Channel, while the three low-order address bits select one of the channel commands, as described in Chapter 2.

A base address is assigned by opening or closing the five most significant switch positions of the S1 switch (S1-4,5,6,7,8) on the Channel Board (see sheet 1 of the Channel Board schematic in Section 3.3). When a switch position is closed (on)(tied to ground) it represents the assignment of a logical 0 address bit. When a position is open (off)(+5V), it represents a logical 1 selection.

The following sketch represents a base address selection of 78_{16}.

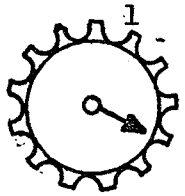


7.5 INTERRUPT LEVEL SELECTION

The user can assign the Diskette Channel's interrupt request line to any one of eight interrupt priority levels (INT0/ - INT7/) by moving the interrupt level select switch (S1) on the Interface Board to the desired position. This nine-position rotary switch is shown on sheet 3 of the Interface Board schematic in Section 4.3. The nine switch positions are associated with the following priority levels:

SWITCH POSITION	INTERRUPT PRIORITY LINE	RELATIVE PRIORITY SBC SYSTEM
1	INT0/	HIGHEST
2	INT1/	↓
3	INT2/	↓
4	INT3/	↓
5	INT4/	↓
6	INT5/	↓
7	INT6/	↓
8	INT7/	LOWEST
9	OFF	X

The following sketch shows the switch setting 3 corresponding to priority line INT2/.



CHAPTER 8
OPERATING CHARACTERISTICS

This chapter provides AC and DC characteristics for all signals not internal to the Diskette controller modules. These signals include the SBC bus signals as well as the drive interface. Refer to the appropriate tables and figures for the desired rating.

8.1 AC CHARACTERISTICS

Tables 8-1 and 8-2 give the AC characteristics for the Diskette Controller boards.

TABLE 8-1

DISKETTE CONTROLLER

SBC BUS AC CHARACTERISTICS

PARAMETER	OVERALL MIN(ns)	OVERALL MAX(ns)	DESCRIPTION	REMARKS
t_{SAS}	50		Address Setup Time to I/O Command	Provided by Host CPU
t_{SDS}	0		Data Setup Time to I/O Command	Provided by Host CPU
t_{SAH}	0		Address Hold Time from I/O Command	Provided by Host CPU
t_{SDHW}	0		Data Hold Time from I/O Command	Provided by Host CPU
t_{SDHR}	25		Read Data Hold Time from I/O Command	Provided by FDCC
t_{ACC}		Bus Timeout	I/O Access Time	Provided by FDCC
t_{XKD}	7		XACK Delay from Read Data	Provided by FDCC
t_{XKO}	30		XACK Hold Time from I/O Command	Provided by FDCC
t_{BCY}	100		Bus Clock Cycle Time	Provided by FPC
t_{BW}	25		Bus Clock Low and High Periods	Provided by FPC
t_{CCY}	100		Common Clock Cycle Time	Provided by FPC
t_{CW}	25		Common Clock Low and High Periods	Provided by FPC
t_{DRQ}		35	Bus Request Delay	
t_{DBY}		65	Bus Busy Turn on Delay	
t_{DBYF}		40	Bus Busy Turn off Delay	
t_{DBPN}	30		Priority Input Setup Time	
t_{DBPO}		20	BPRO/ Serial Delay from BPRN/	
t_{DB}	50		Busy to Address/Data Delay	
$t_{AS,DS}$	60		Address/Data Setup to Command	
t_{XKCO}	35	125	XACK to Command Turn off	
t_{AH}	115		Address Hold Time	
t_{DHW}	125		Data Hold Time	
t_{DHR}	0		Read Data Hold Time	Provided by Slave
t_{BS}	200		Bus Sample Delay Time	

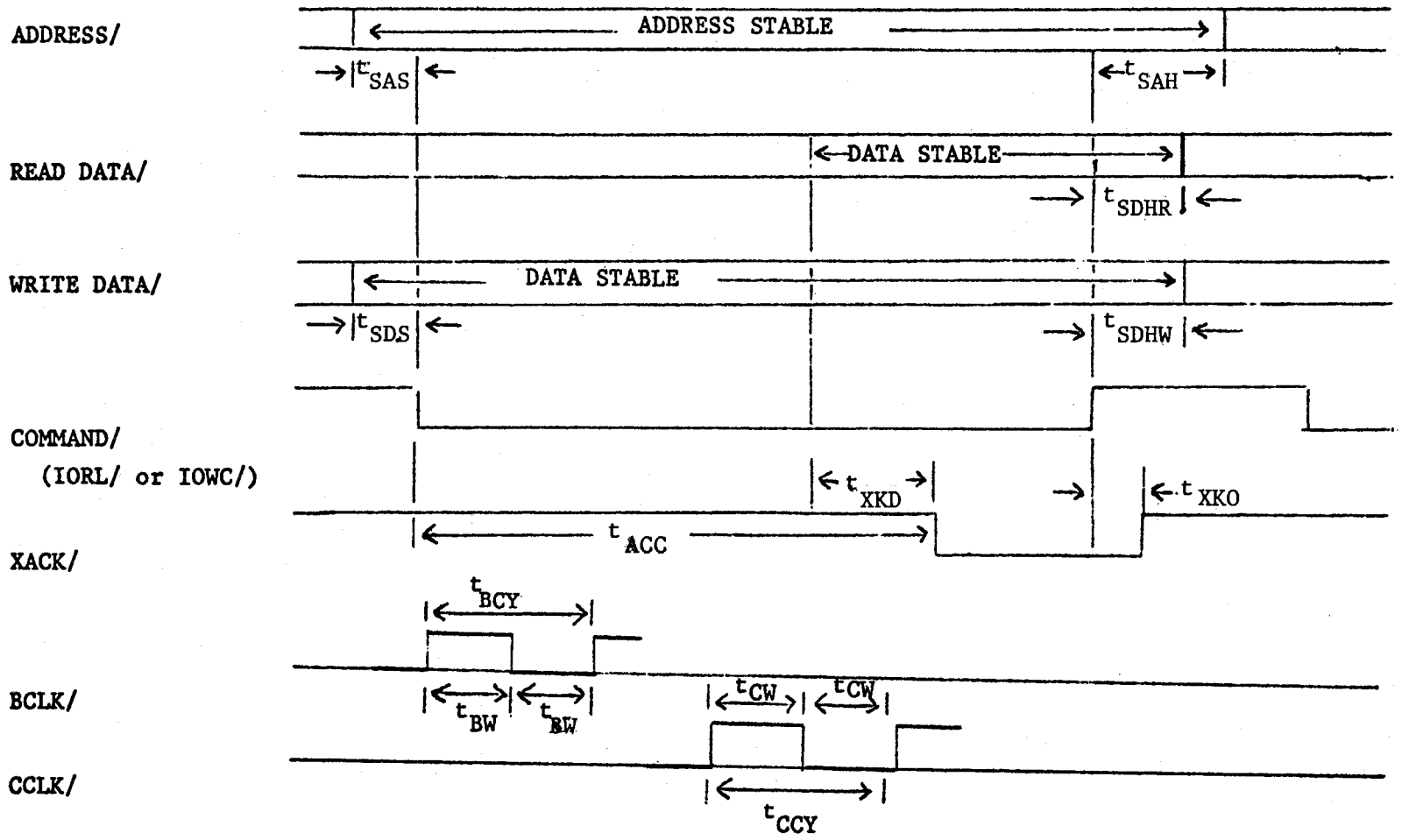


FIGURE 8-1

SLAVE COMMAND TIMING - FDCC

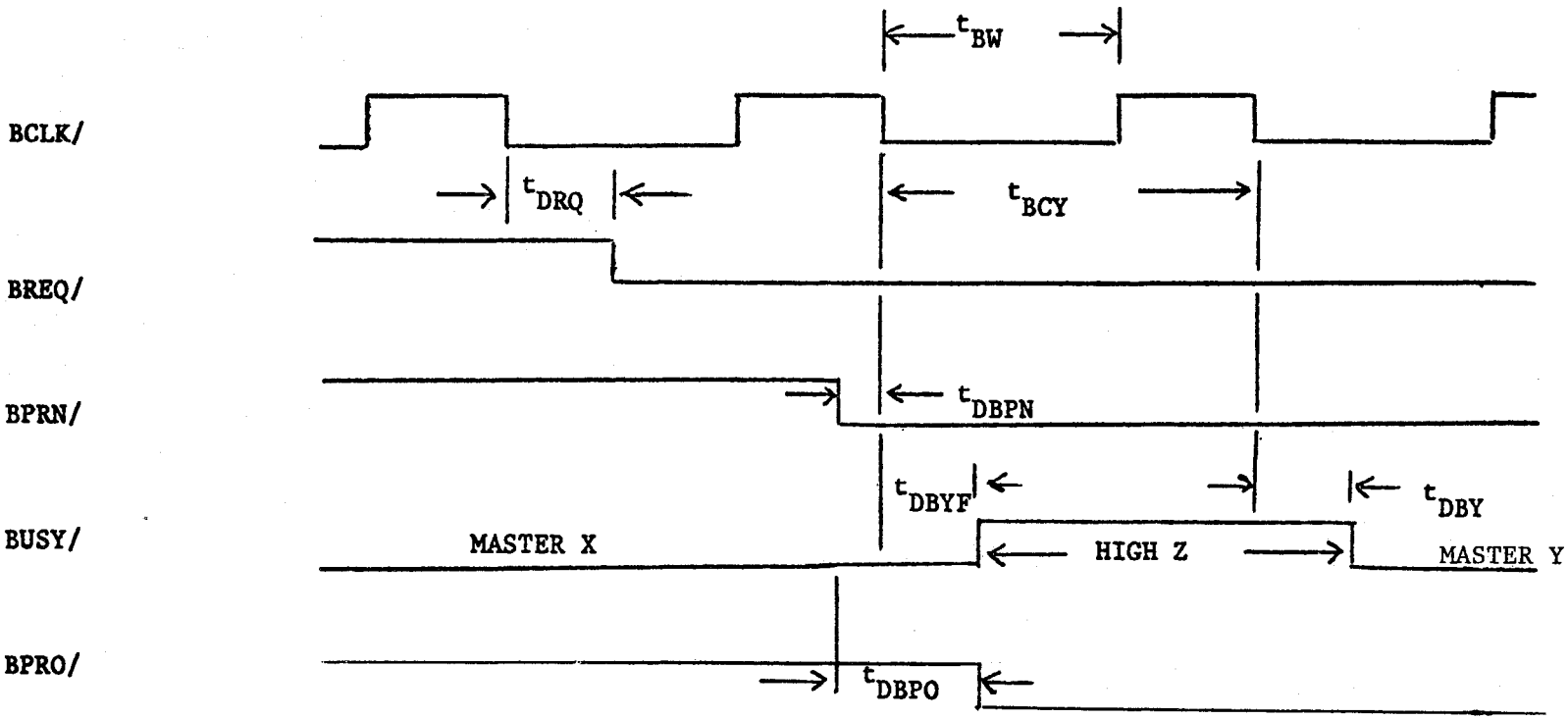


FIGURE 8-2

BUS EXCHANGE TIMING

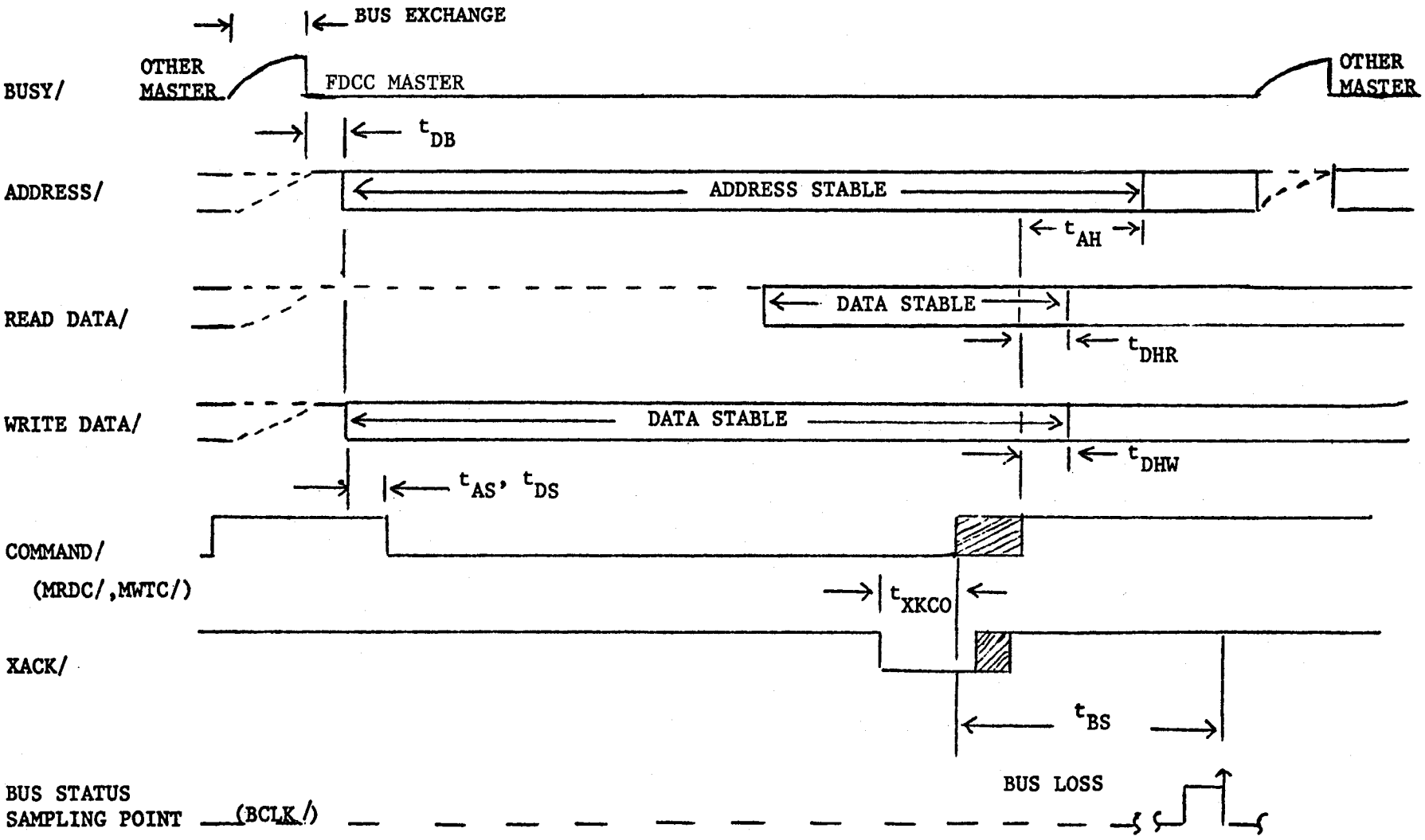


FIGURE 8-3

MASTER COMMAND TIMING

TABLE 8-2

DISKETTE CONTROLLER

DRIVE INTERFACE AC CHARACTERISTICS

PARAMETER	OVERALL MIN	OVERALL TYP	OVERALL MAX	DESCRIPTION	REMARKS
t_{SDIR}		60 msec		Select to Direction Set	
t_{SSTP}		60 msec		Select to Step Pulse	
t_{SCYS}	8 msec	10 msec	12 msec	Step Cycle Time	
t_{SPW}		100 µsec		Step Pulse Width	
t_{LSET}		60 msec		Settling Time - Select to Data	
t_{SSET}		20 msec		Settling Time - Step to Data	
t_{CLCL}		4 µsec		Clock-to-Clock Spacing (Rd)	Subject to Rotational Tolerance
t_{RCPW}	200 nsec	250 nsec	300 nsec	Clock Pulse Width (Rd)	
t_{DTDT}		4 µsec		Data-to-Data Spacing (Rd)	Subject to Rotational Tolerance
t_{RDPW}	200 nsec	250 nsec	300 nsec	Data Pulse Width (Rd)	
t_{CLDT}		2 µsec		Clock-to-Data Spacing (Rd)	Subject to Rotational Tolerance
t_{CELL}		4 µsec		Clock-to-Clock Spacing (Wt)	
t_{DAT}		4 µsec		Clock-to-Data Spacing (Wt)	
t_{WCPW}		250 nsec		Clock Pulse Width (Wt)	
t_{WDPW}		250 nsec		Data Pulse Width (Wt)	
t_{NXPW}		1.5 msec		Index Pulse Width	
t_{NXCY}		166.7 msec		Index Cycle Time	
t_{WFPW}	2 µsec		2.8 µsec	Write Fault Reset Pulse Width	

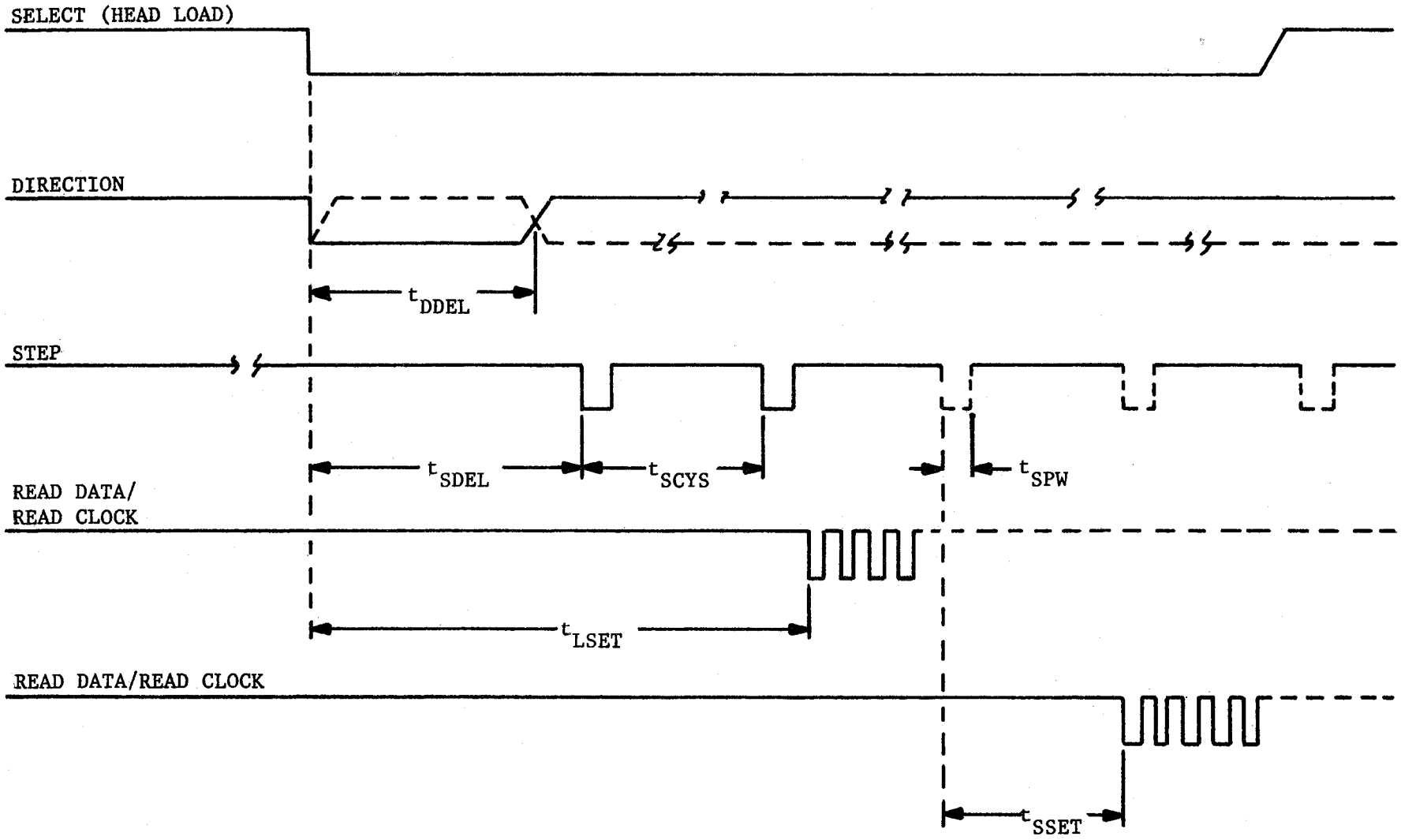


FIGURE 8-4

TIMING DIAGRAM: STEP PULSE AND SETTLING TIME

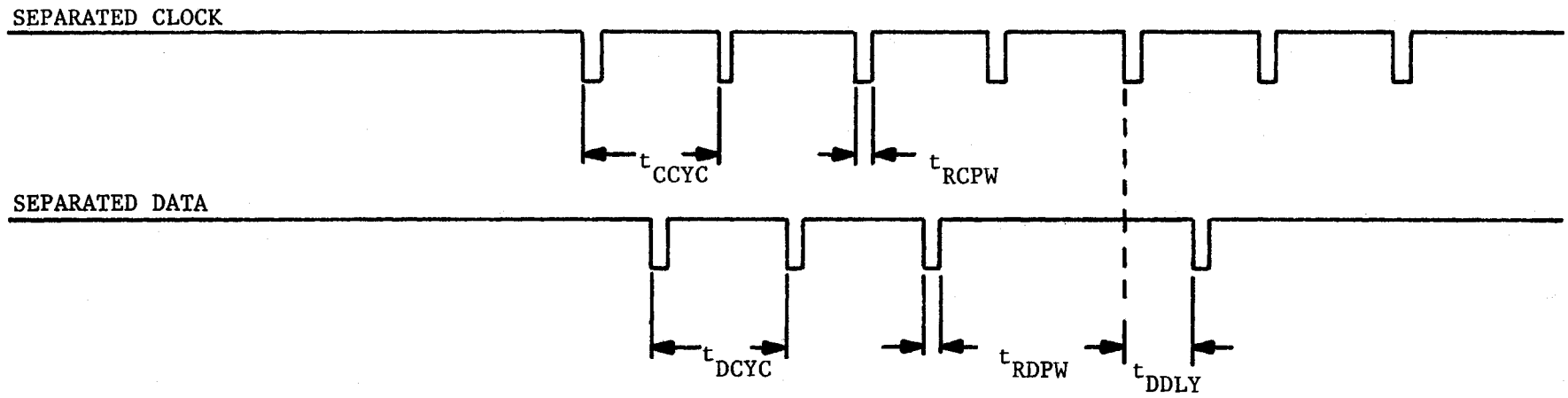


FIGURE 8-5

READ TIMING

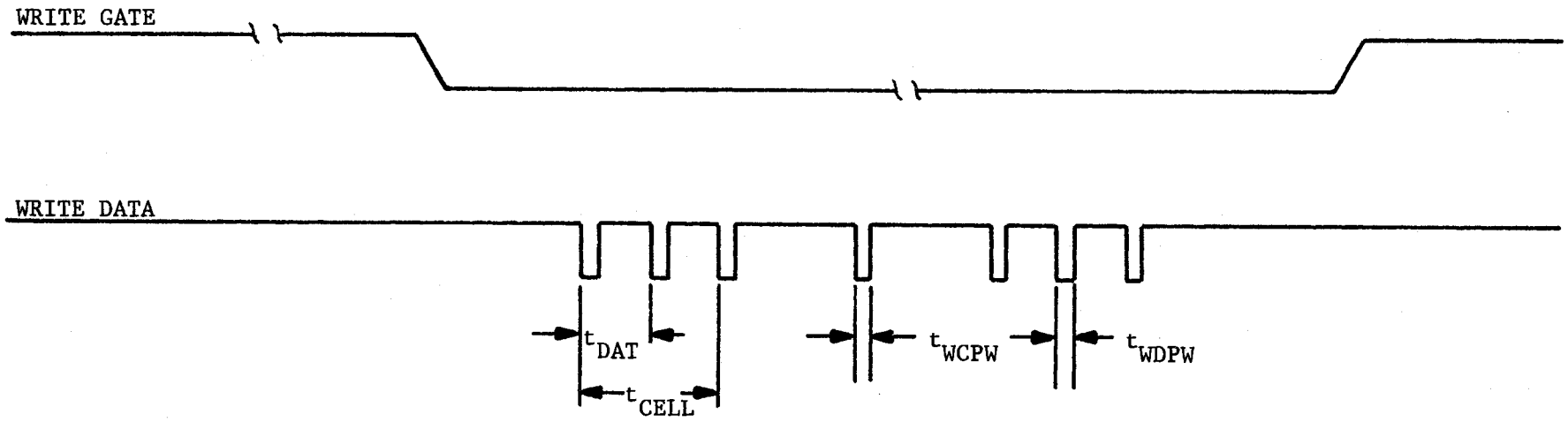


FIGURE 8-6

WRITE TIMING

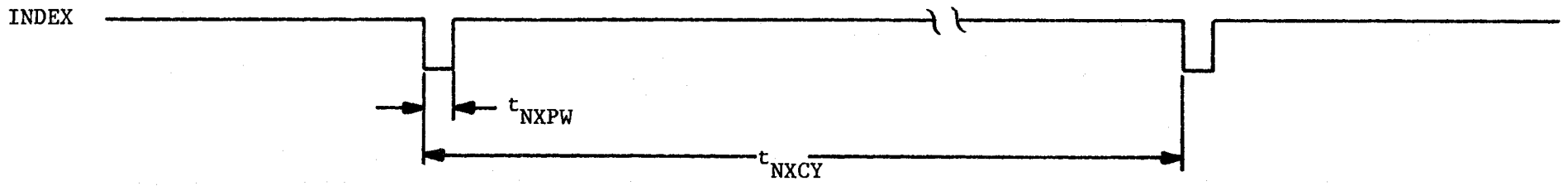


FIGURE 8-7

INDEX TIMING

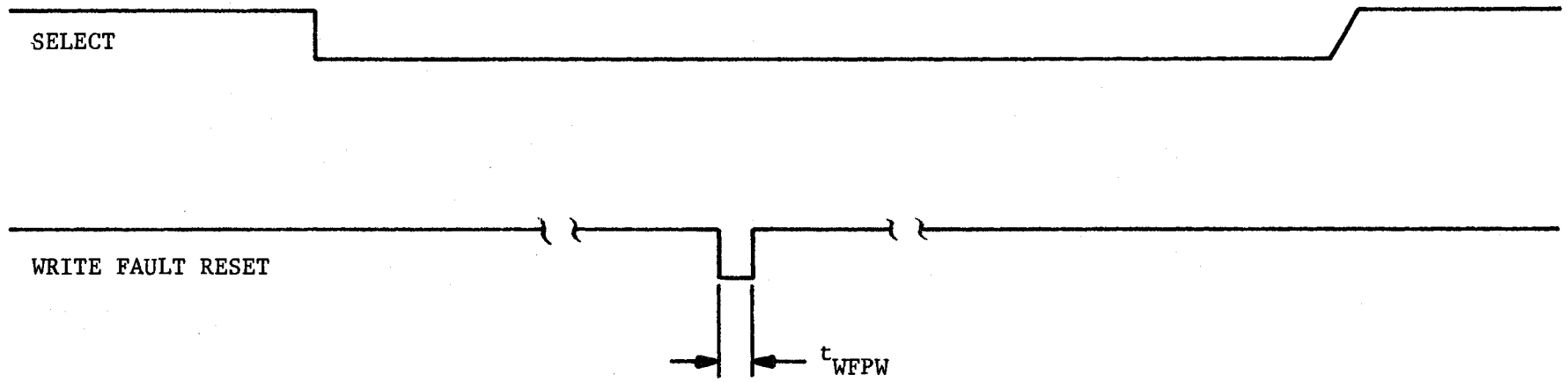


FIGURE 8-8

WRITE FAULT RESET TIMING

8.2 DC CHARACTERISTICS

The following two tables (8-3 and 8-4) give the DC characteristics for the Channel and Interface Boards. Both the SBC bus and drive interface are included. The values are derived from manufacturers' specifications and calculated values if passive loading exists. Capacitance values are approximations.

Power requirements for the boards are as follows:

	Vcc(+5VDC ±5%)	
	TYP	MAX
CHANNEL BOARD	3.75A	5.0A
INTERFACE BOARD	1.5A	2.5A
TOTAL	5.25A	7.5A

TABLE 8-3

DISKETTE CONTROLLER

DC CHARACTERISTICS (SBC)

SIGNAL	SYMBOL	PARAMETER DESCRIPTION	TEST CONDITION	PARAMETER MIN	MAX	UNITS
ADRF/-ADR8/	V_{OL}	Output Low Voltage	I_{OL}=10mA		.45	V
	V_{OH}	Output High Voltage	I_{OH}=-1mA	2.4		V
	C	Capacitive Load			15	pf
ADR7/-ADRϕ/	V_{OL}	Output Low Voltage	I_{OL}=15mA		.45	V
	V_{OH}	Output High Voltage	I_{OH}=-1mA	3.65		V
	V_{IL}	Input Low Voltage			.8	V
	V_{IH}	Input High Voltage		2		V
	I_{IL}	Input Current at V_{IL}	V_{IL}= .4V		-1.6	mA
	I_{IH}	Input Current at V_{IH}	V_{IH}=2.4V		40	µA
	C	Capacitive Load			15	pf
DATF/-DAT4/	V_{OL}	Output Low Voltage	I_{OL}=15mA		.45	V
	V_{OH}	Output High Voltage	I_{OH}=-1mA	3.65		V
	V_{IL}	Input Low Voltage			.85	V
	V_{IH}	Input High Voltage		2		V
	I_{IL}	Input Current at V_{IL}	V_{IL}= .45V		-.25	mA
	I_{IH}	Input Current at V_{IH}	V_{IH}=5.25V		10	µA
	C	Capacitive Load			15	pf
DAT3/-DATϕ/	V_{OL}	Output Low Voltage	I_{OL}=15mA		.45	V
	V_{OH}	Output High Voltage	I_{OH}=-1mA	3.65		V
	V_{IL}	Input Low Voltage			.85	V
	V_{IH}	Input High Voltage		2		V
	I_{IL}	Input Current at V_{IL}	V_{IL}= .45V		-.25	mA
	I_{IH}	Input Current at V_{IH}	V_{IH}=5.25V		10	µA
	C	Capacitive Load			15	pf
BCLK/	V_{IL}	Input Low Voltage			.8	V
CCLK/	V_{IH}	Input High Voltage		2		V
	I_{IL}	Input Current at V_{IL}	V_{IL}= .4V		-2	mA
	I_{IH}	Input Current at V_{IH}	V_{IH}=2.4V		50	µA
	C	Capacitive Load			15	pf

TABLE 8-3 - Continued

SIGNAL	SYMBOL	PARAMETER DESCRIPTION	TEST CONDITION	PARAMETER MIN	MAX	UNITS
BPRN/	V_{IL}	Input Low Voltage			.8	V
	V_{IH}	Input High Voltage		2		V
	I_{IL}	Input Current at V_{IL}	V_{IL}= .4V		-3.2	mA
	I_{IH}	Input Current at V_{IH}	V_{IH}=2.4V		80	µA
	C	Capacitive Load			15	pf
IORC/,IOWC/	V_{IL}	Input Low Voltage			.8	V
INIT/	V_{IH}	Input High Voltage		2		V
	I_{IL}	Input Current at V_{IL}	V_{IL}= .4V		-1.6	mA
	I_{IH}	Input Current at V_{IH}	V_{IH}=2.4V		40	µA
	C	Capacitive Load			15	pf
XACK/	V_{OL}	Output Low Voltage	I_{OL}=16mA		.4	V
	V_{OH}	Output High Voltage	I_{OH}=-5.2mA	2.4		V
	V_{IL}	Input Low Voltage			.8	V
	V_{IH}	Input High Voltage		2		V
	I_{IL}	Input Current at V_{IL}	V_{IL}= .4V		-1.6	mA
	I_{IL}	Input Current at V_{IH}	V_{IH}=2.4V		40	µA
	I_{LL}	Input Leakage Low	High Z V_O=2.4V		-40	µA
	I_{LH}	Input Leakage High	High Z V_O= .4V		40	µA
	C	Capacitive Load			15	pf
BPRO/	V_{OL}	Output Low Voltage	I_{OL}=16mA		.4	V
	V_{OH}	Output High Voltage	I_{OH}=-800µA	2.4		V
	C	Capacitive Load			15	pf
BREQ/	V_{OL}	Output Low Voltage	I_{OL}=20mA		.4	V
	V_{OH}	Output High Voltage	I_{OH}=-500µA	2.4		V
	C	Capacitive Load			15	pf
MRDC/,MWTC/	V_{OL}	Output Low Voltage	I_{OL}=32mA		.4	V
	V_{OH}	Output High Voltage	I_{OH}=-5.2mA	2.4		V
	C	Capacitive Load			15	pf

TABLE 8-3 - Continued

SIGNAL	SYMBOL	PARAMETER DESCRIPTION	TEST CONDITION	PARAMETER		
				MIN	MAX	UNITS
BUSY/	V_{OL}	Output Low Voltage	I_{OL}=16mA		.4	V
	V_{OH}	Output High Voltage	I_{OH}=-5.2mA	2.4		V
	C	Capacitive Load			15	pf
INT7/-INTϕ/	V_{OL}	Output Low Voltage	I_{OL}=16mA		.4	V
	I_{OH}	Output High Leakage	Open Collector Off		250	µA
	C	Capacitive Load	V_{OH}=5.5V		15	pf

TABLE 8-4

DISKETTE CONTROLLER

DC CHARACTERISTICS (DRIVE/DISPLAY INTERFACE)

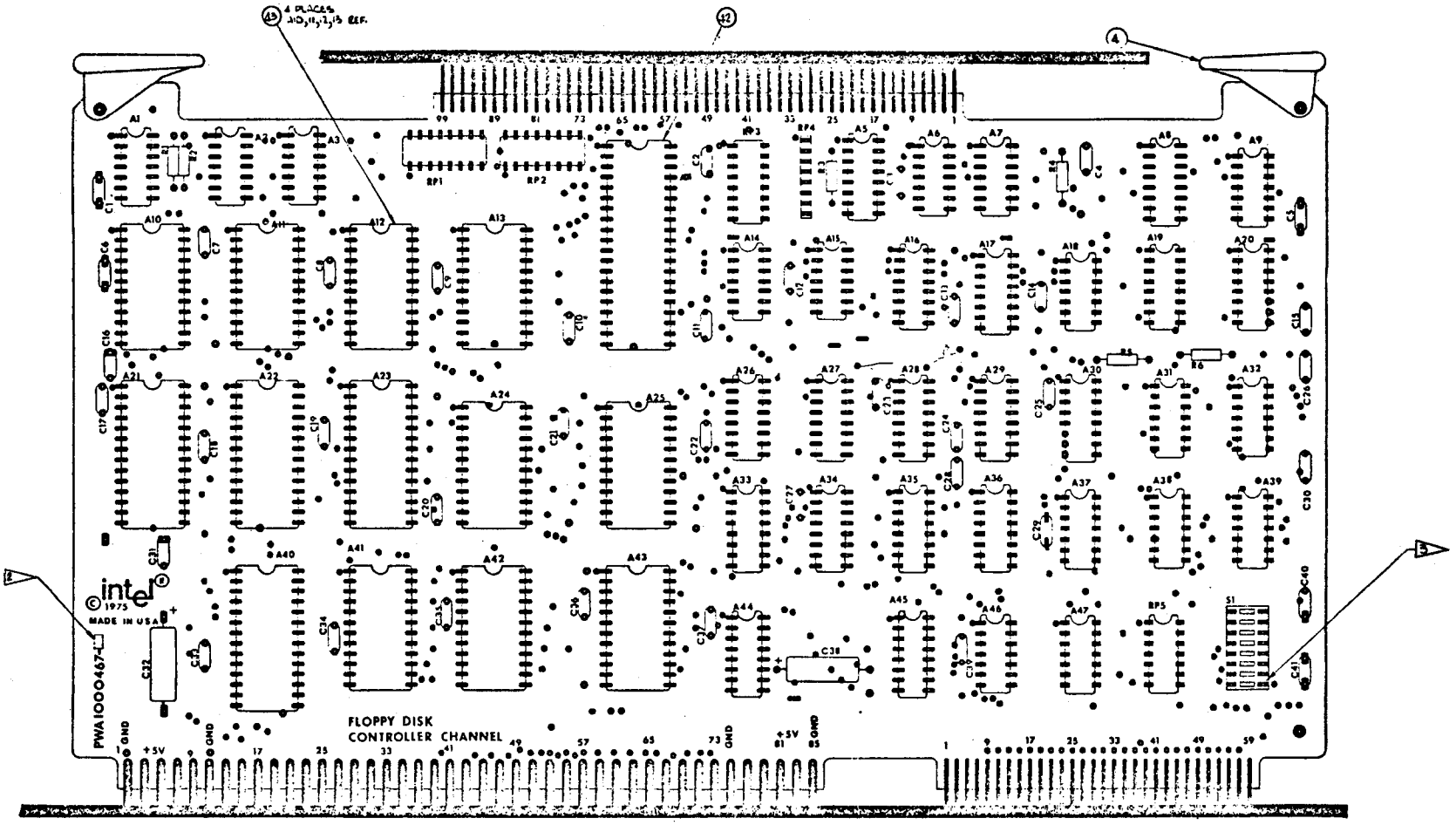
SIGNAL	SYMBOL	PARAMETER DESCRIPTION	TEST CONDITION	PARAMETER MIN	MAX	UNITS
SELϕ/	V_{OL}	Output Low Voltage	I_{OL}=48mA		.4	V
SEL1/	I_{OH}	Output High Leakage	Open Collector Off V_{OH}=5.5V		250	µA
WRT DAT/	C	Capacitive Load			15	pf
WRT GT/						
STEP/						
DIR/						
WRT FLT RESET/						
TRACK GT 43/						
READY ϕ/	V_{IL}	Input Low Voltage			.8	V
READY 1/	V_{IH}	Input High Voltage		3.0		V
SEP DATA/	I_{IL}	Input Current at V_{IL}	V_{IL}= .8		-16	mA (1)
SEP CLOCK/	I_{IH}	Input Current at V_{IH}	V_{IH}=3.0		260	µA (2)
TRACK ϕ/	C	Capacitive Load			15	pf
INDEX/						
WRT FLT/						
WPROT/						
LEDϕ/	V_{OL}	Output Low Voltage	I_{OL}=16mA		.4	V
LED1/	V_{OH}	Output High Voltage	I_{OH}=-400µA	2.4		V
	C	Capacitive Load			15	pf

(1) Includes 9.1mA due to 220/330Ω pull-up.

(2) Includes 250mA due to 220/330Ω pull-up.

APPENDIX A

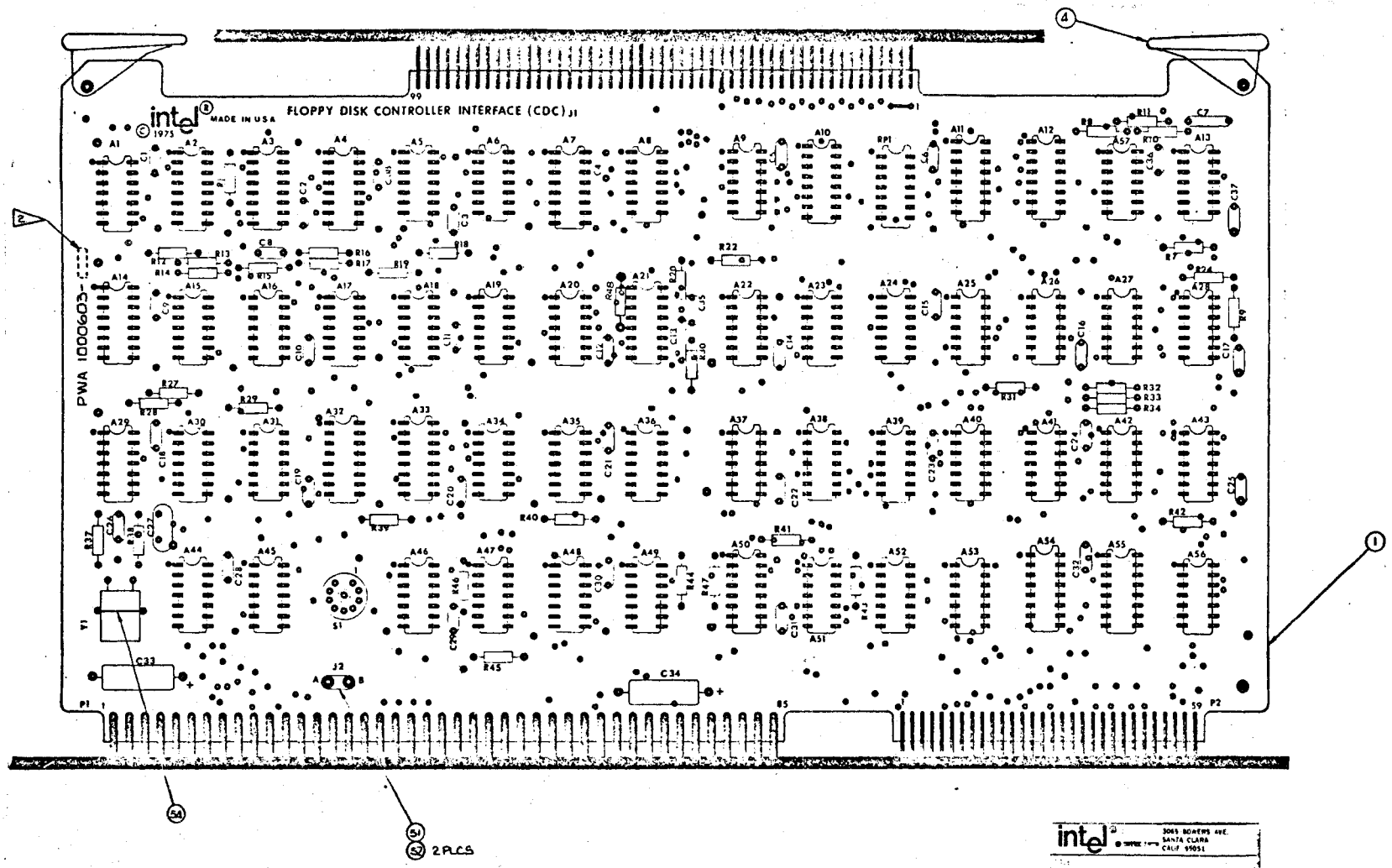
SBC 201, SBC 211, SBC 212
ASSEMBLY DRAWINGS



NOTE: UNLESS OTHERWISE SPECIFIED

1. ASSEMBLY NO. IS 1000467-01.
2. MARK DASH NO. IN POSITION SHOWN.
3. INSTALL SWITCH WITH SLIDE POSITION 1 TO BOTTOM OF BOARD.

DESCRIPTION
intel 3065 BOWERS AVE SANTA CLARA CALIF 95051
PRINTED WIRING ASSEMBLY
FLOPPY DISK CONTROLLER CHANNEL
D 410 1000467 F

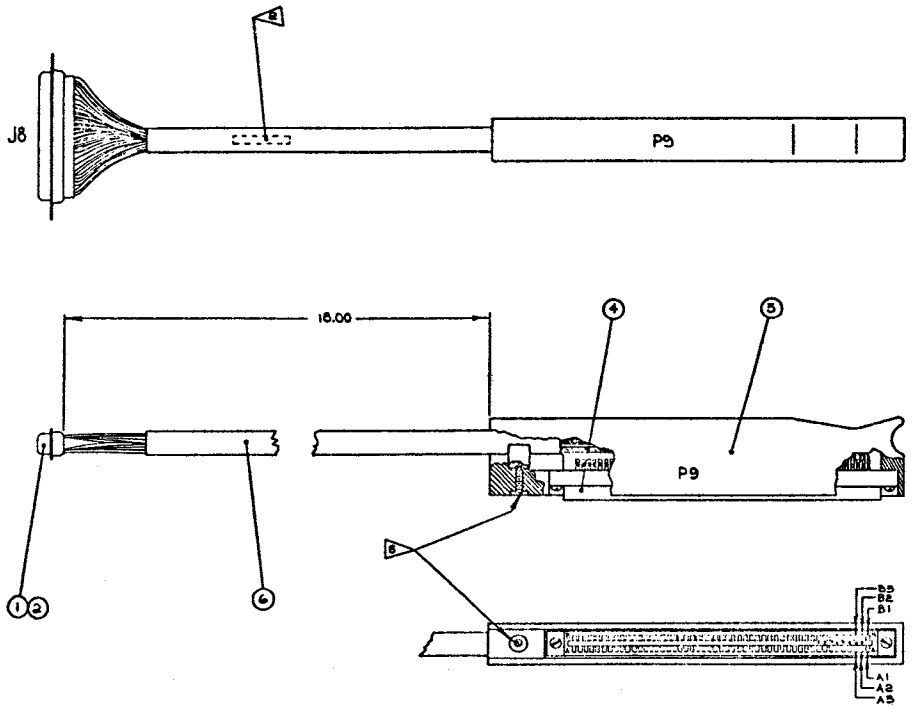


NOTE: UNLESS OTHERWISE SPECIFIED
1. ASSEMBLY NO. IS 1000603-01
2 MARK DASH NO. IN POSITION SHOWN.

intel 3065 BOWERS AVE. SANTA CLARA CALIF. 95051

PRINTED WIRING ASSEMBLY
FLOPPY DISK CONTROLLER INTERFACE
CDC

D 410 1000603 D

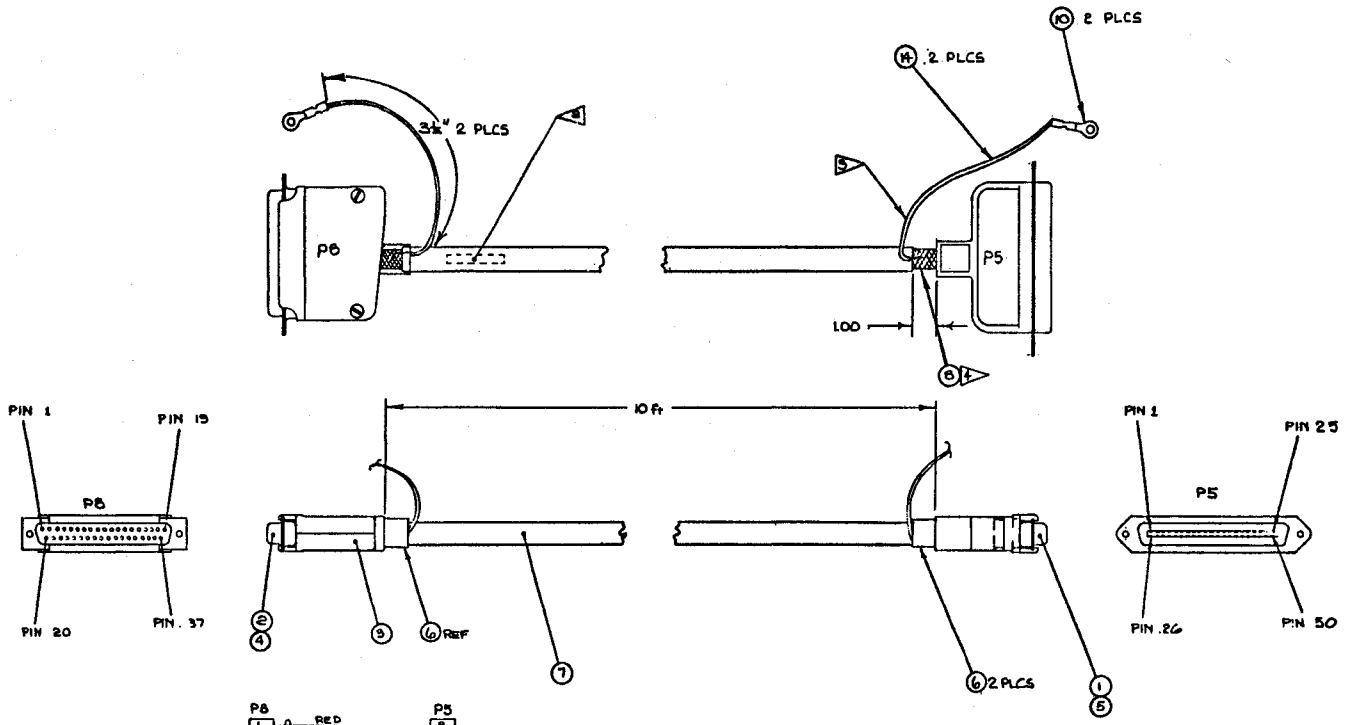


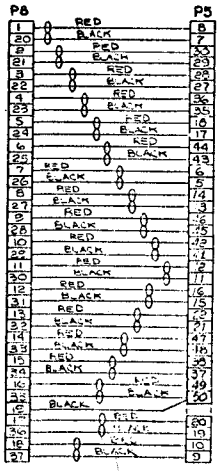
J8		P8
1	RED	A1
20	BLACK	B1
2	RED	A3
21	BLACK	B3
3	RED	B4
22	BLACK	A4
4	RED	B6
23	BLACK	A6
5	RED	B2
24	BLACK	A2
6	RED	A7
25	BLACK	B7
7	RED	A8
26	BLACK	B8
8	RED	B10
27	BLACK	A10
9	RED	A11
28	BLACK	B11
10	RED	A5
29	BLACK	B5
11	RED	C12
30	BLACK	A12
12	RED	B13
31	BLACK	A14
13	RED	A15
32	BLACK	B15
14	RED	A17
33	BLACK	B17
15	RED	B18
34	BLACK	A18
16	RED	B16
35	BLACK	A16
17	RED	A19
36	BLACK	B19
18	RED	A13
37	BLACK	B13
19	RED	B20

NOTES:
UNLESS OTHERWISE SPECIFIED,

1. ASSY NO IS 4000522-01.
2. INK STAMP ASSY NO AND REV LTR IN CONTRASTING INK USING MIN .12 HIGH CHARACTERS.
3. THIS SCREW CONTROLS THE DEGREE OF STRAIN RELIEF. BRING VINYL TUBING PAST THIS SPOT INTO THE CONN HOOD CHAMBER AS SHOWN.

intel © INTEL 1973	3065 BOWERS AVE. SANTA CLARA CALIF. 95051		
TITLE: FLOPPY DISK CONTROLLER CABLE			
SIZE: D	DEPT: 410	DRAWING NO.: 4000522	REV: B1

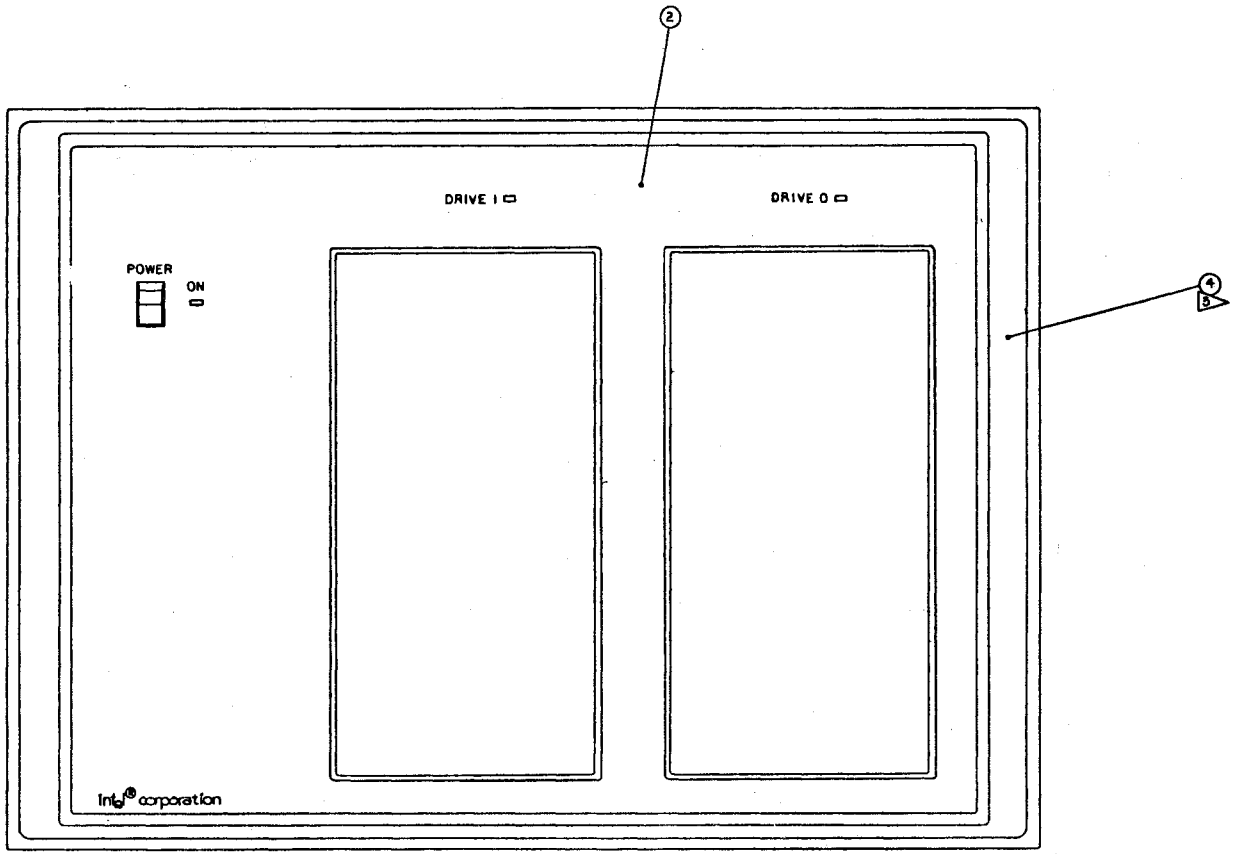




NOTES:
UNLESS OTHERWISE SPECIFIED,
1. ASSY NO IS 4000521-01.
2. INK STAMP ASSY NO AND REV LTR IN CONTRASTING COLOR USING MIN .12 HIGH CHARACTERS.
3. ATTACH GND WIRE (ITEM 14) TO MESH (ITEM 8) BOTH ENDS.
4. ITEM 6 NOT SHOWN FOR CLARITY.

intel 3065 BOWERS AVE. SANTA CLARA CALIF. 95051

TITLE			
FLOPPY DISK PERIPHERAL CABLE			
SIZE	DEPT	DRAWING NO.	REV
D	410	4000521	E



FRONT VIEW

NOTES:
UNLESS OTHERWISE SPECIFIED,

1. ASSY NO IS 4000533-XX.

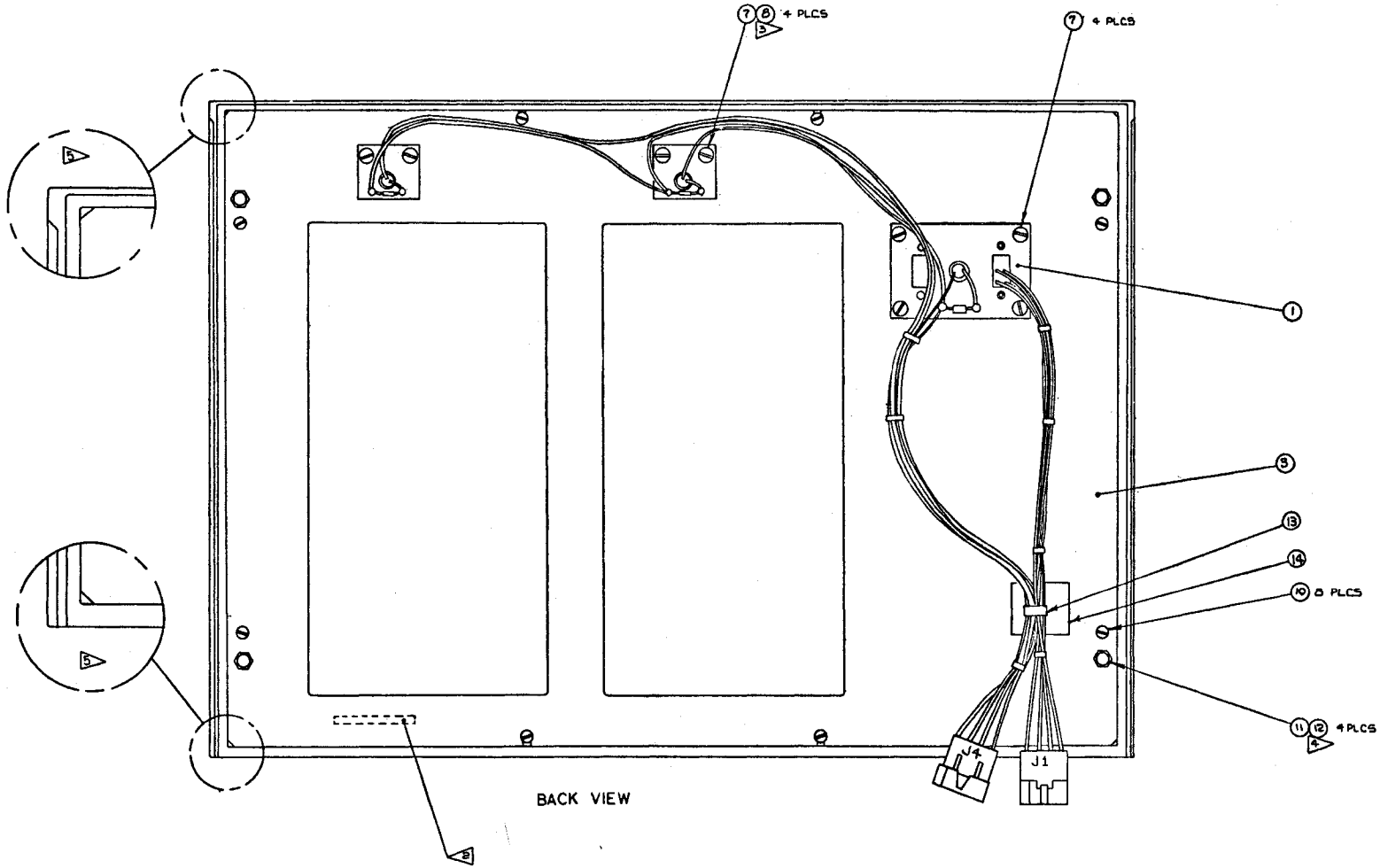
4 INK STAMP ASSY NO AND REV LTR, IN CONTRASTING COLOR, CHARACTERS TO BE MIN .12 HIGH.

5 PLACE WASHER (ITEM 8) BETWEEN PLATE AND SUB-PANEL.

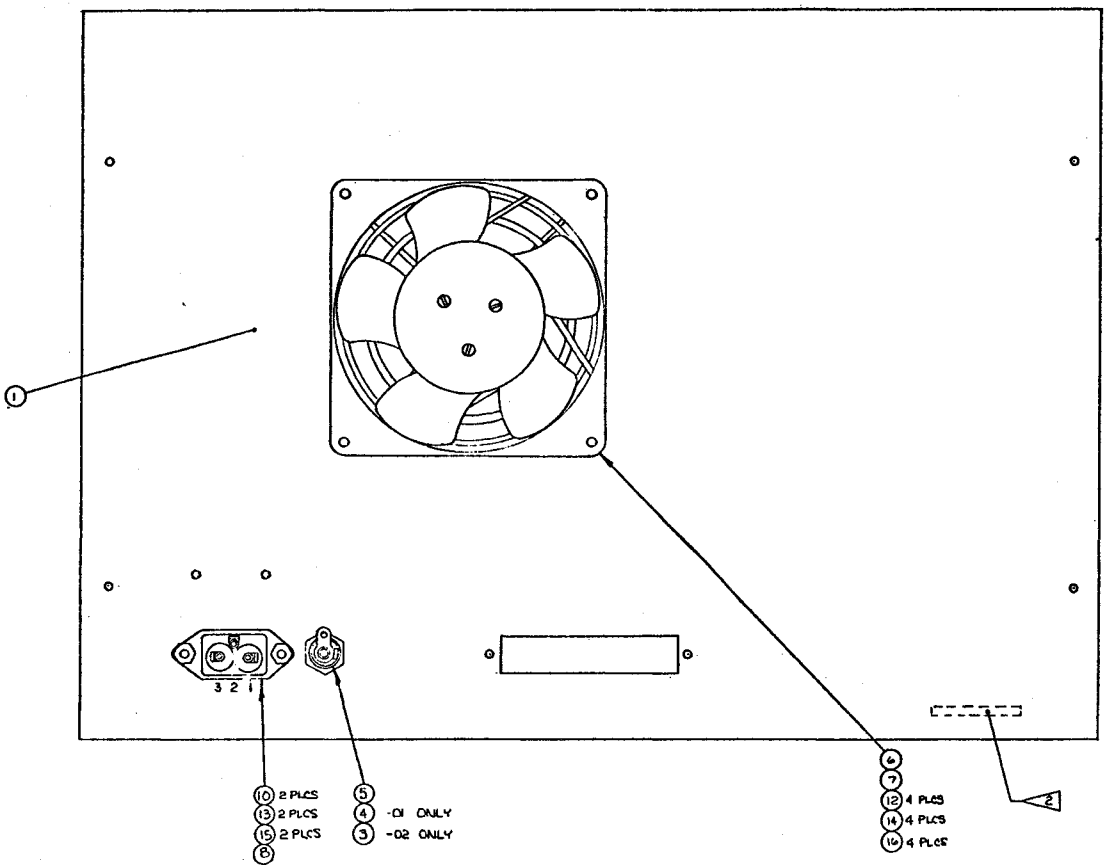
4 ATTACH BALL STUD (ITEM 11) TO SUB-PANEL (ITEM 3) WITH SCREW (ITEM 12) BEFORE ADHERING OVERLAY (ITEM 2) TO SUB-PANEL. OVERLAY WILL COVER AND CONCEAL THE SCREW USED TO ATTACH THE BALL STUD.

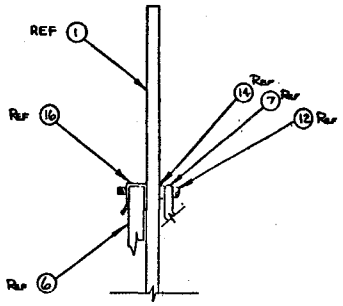
5 ENSURE BEZEL IS INSTALLED AS SHOWN ON SHEET 2.

intel	3065 BOWERS AVE. SANTA CLARA CALIF. 95051		
TITLE: FRONT PANEL ASSEMBLY			
SIZE: D	DEPT: 410	DRAWING NO.: 4000533	REV: B



SCALE: 1/1 SHEET 2 OF 2	SIZE D	DEPT 410	DRAWING NO. A000533	REV B

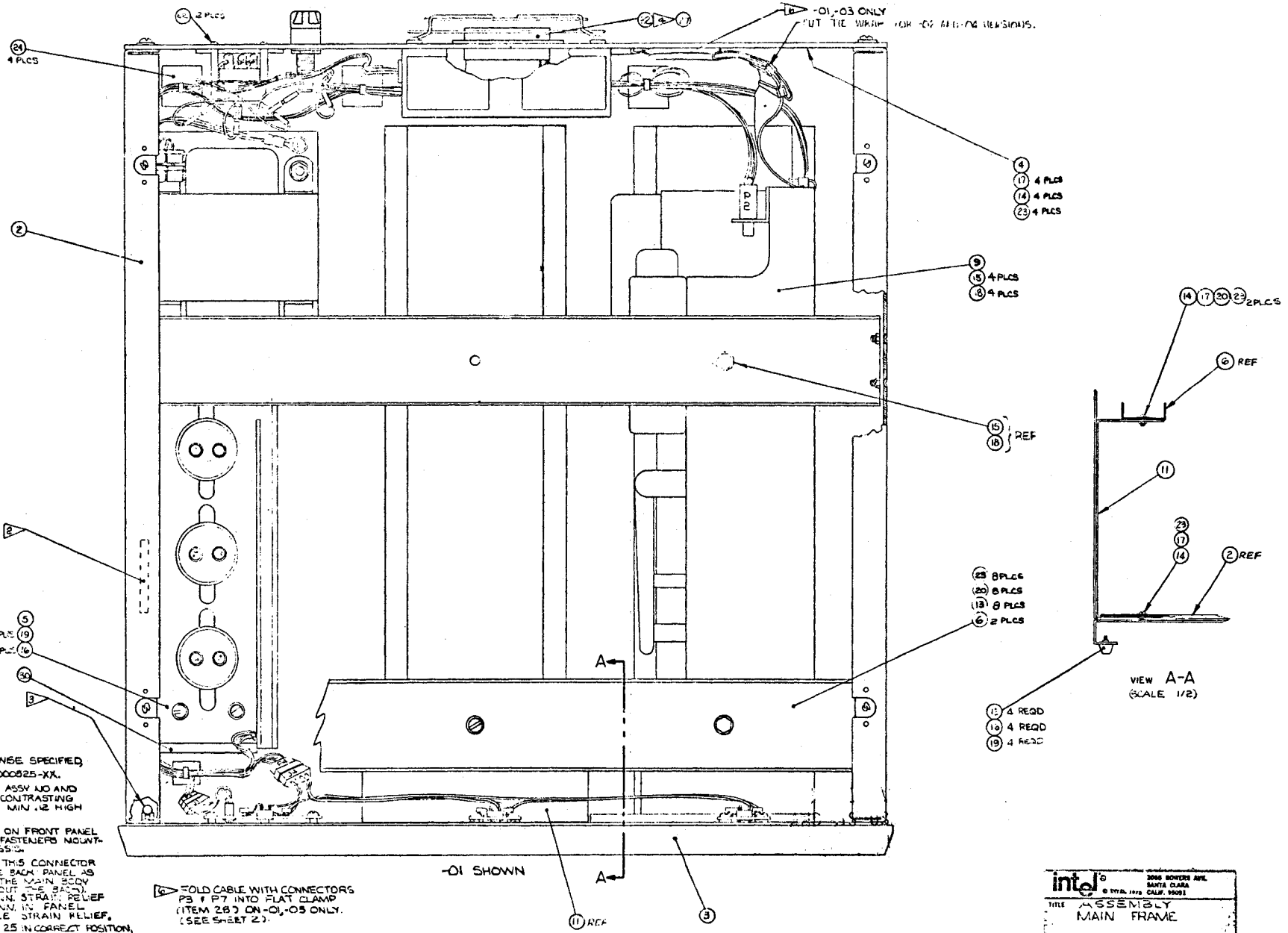




SIDE VIEW
(Not To Scale)

NOTES:
UNLESS OTHERWISE SPECIFIED,
1. ASSY NO IS 4000531-XX.
A INK STAMP ASSY NO AND REV LETTER IN CONTRASTING COLOR USING MIN .12 HIGH CHARACTERS.
B INK STAMP TERMINAL POSITIONS FOR LINE FILTER (ITEM 5) AS SHOWN USING MIN .12 HIGH CHARACTERS IN CONTRASTING COLOR.

intel	3065 EDCLEN AVE. SANTA CLARA CALIF. 95051		
TITLE: REAR PANEL ASSEMBLY			
SIZE: D	DEPT: 110	DRAWING NO: 4000531	REV: F



NOTES:
UNLESS OTHERWISE SPECIFIED,
1. ASSY NO IS 4000825-XX.

2. INK STAMP ASSY NO AND REV LTR IN CONTRASTING COLOR USING MIN .12 HIGH CHARACTERS.

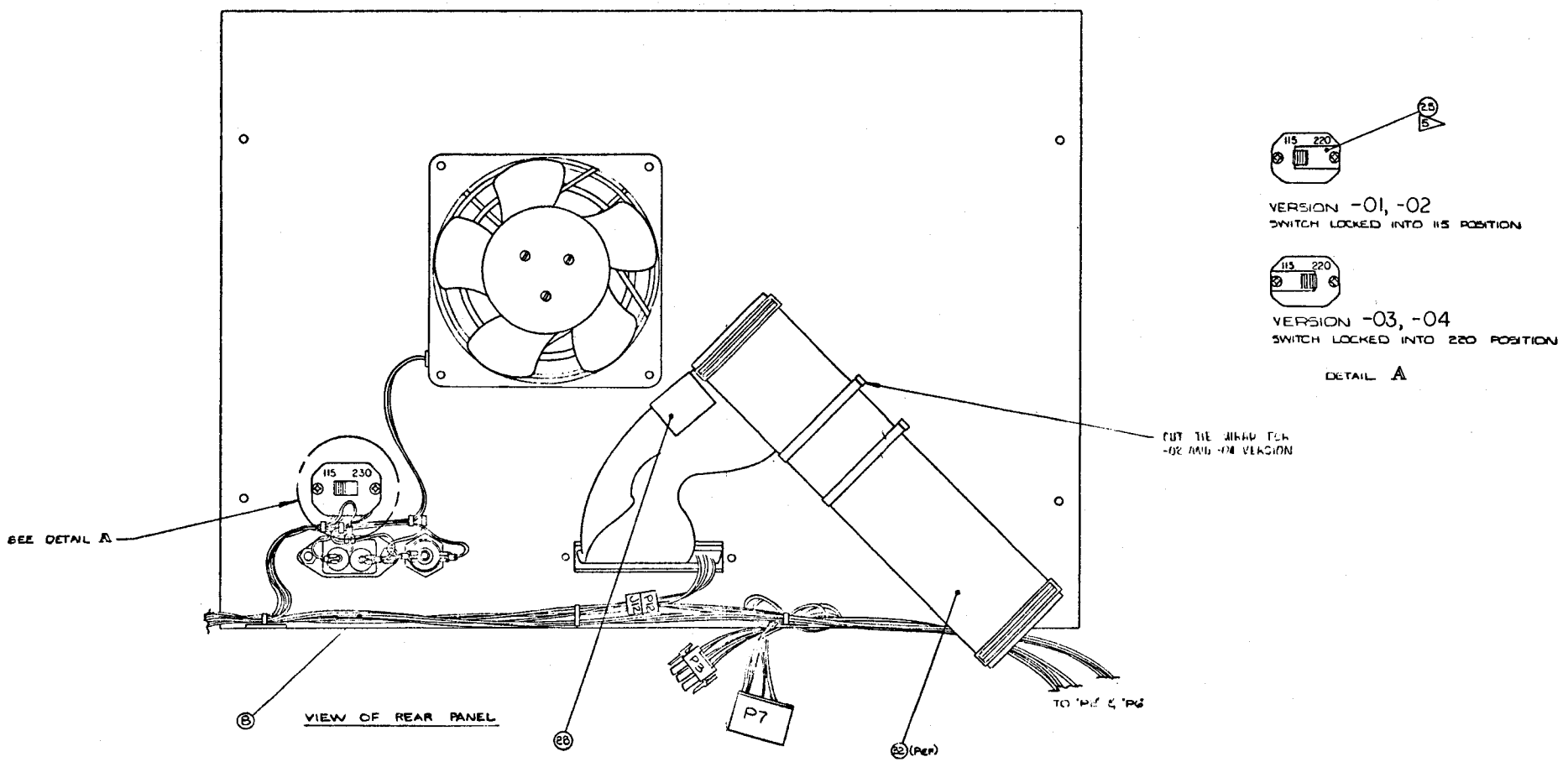
3. BALL STUDS ON FRONT PANEL "SNAP" INTO FASTENERS MOUNTED ON CHASSIS.

4. TO MOUNT THIS CONNECTOR (J5) INTO THE BACK PANEL AS SHOWN (THE MAIN BODY PROTRUDES OUT THE BACK), REMOVE CONN. STRAIN RELIEF. INSTALL CONN. IN PANEL REASSEMBLE STRAIN RELIEF.

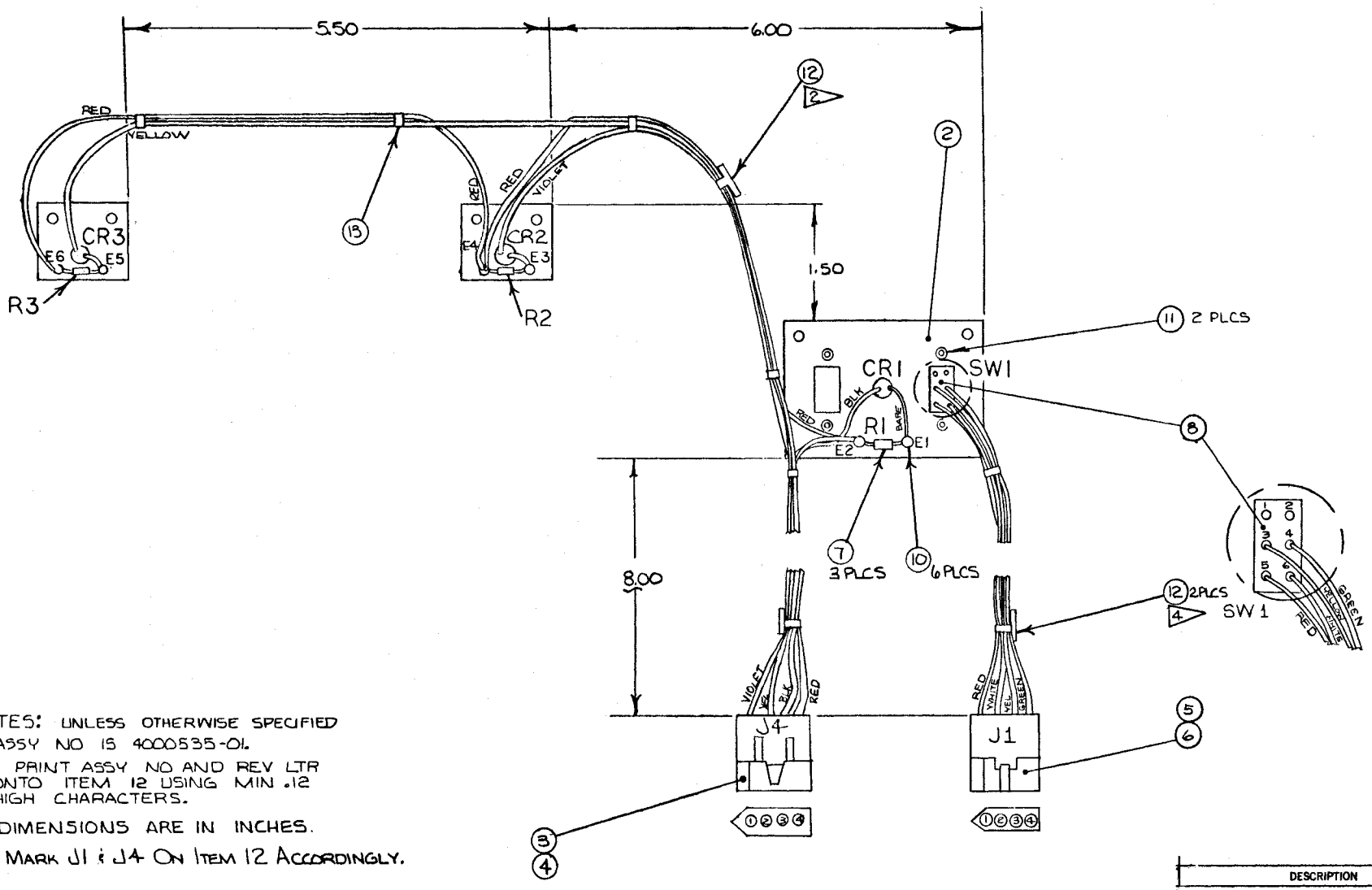
5. INSTALL ITEM 25 IN CORRECT POSITION, FOR THE APPROPRIATE VERSION, AS SHOWN.

6. FOLD CABLE WITH CONNECTORS P3 & P7 INTO FLAT CLAMP (ITEM 28) ON -01,-05 ONLY. (SEE SHEET 2).

intel © 1978	3065 BOWERS AVE. SANTA CLARA CALIF. 95051		
TITLE ASSEMBLY MAIN FRAME			
SIZE D	DEPT 410	DRAWING NO. 4000525	REV E



SCALE: FULL	SIZE	DEPT	DRAWING NO.	REV
SHEET 2 OF 3	D	410	4000525	E



NOTES: UNLESS OTHERWISE SPECIFIED

1. ASSY NO IS 4000535-01.
2. PRINT ASSY NO AND REV LTR ONTO ITEM 12 USING MIN .12 HIGH CHARACTERS.
3. DIMENSIONS ARE IN INCHES.
4. MARK J1 & J4 ON ITEM 12 ACCORDINGLY.

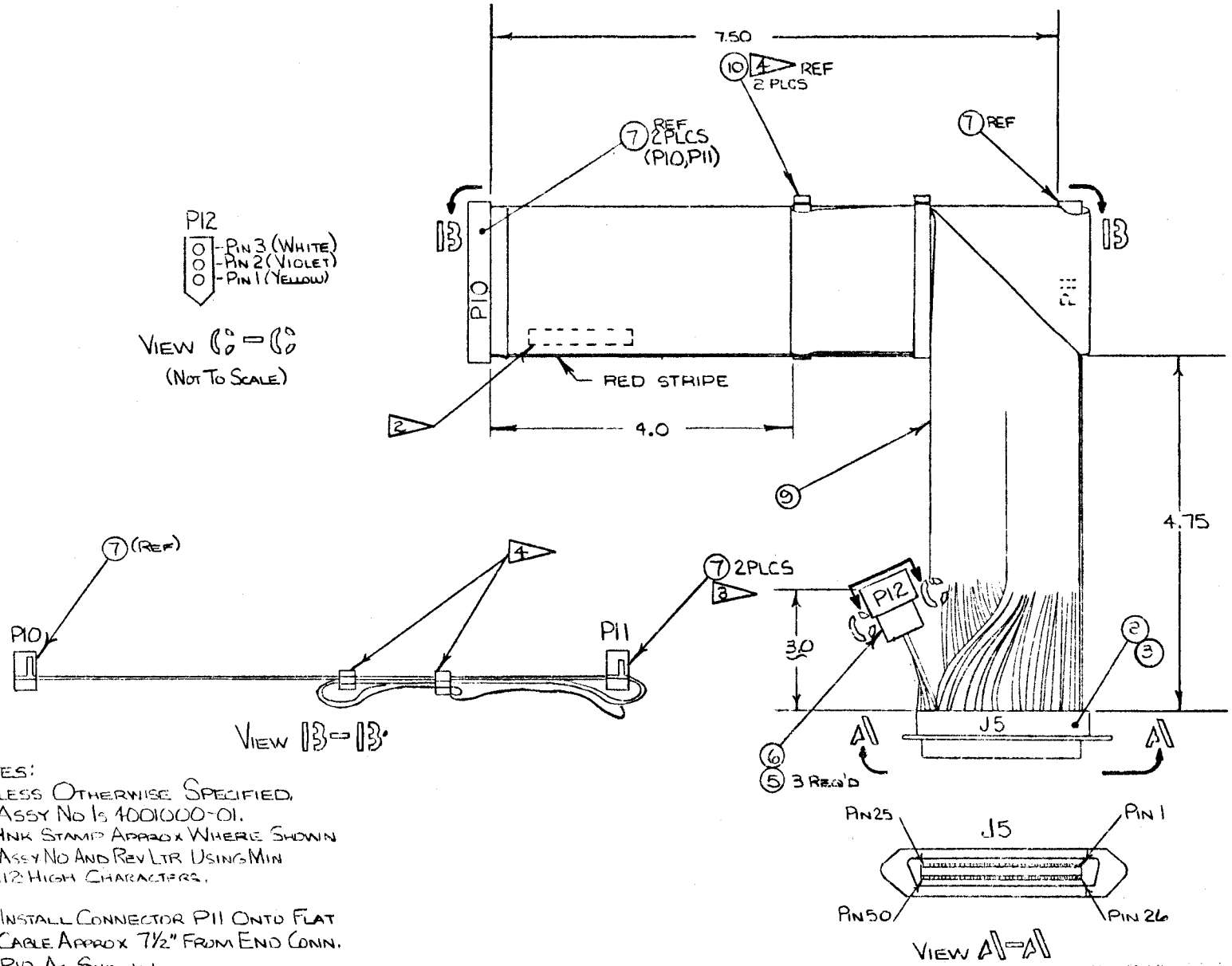
DESCRIPTION

PARTS LIST

intel © INTEL 1978
3065 BOWERS AVE.
SANTA CLARA
CALIF. 95051

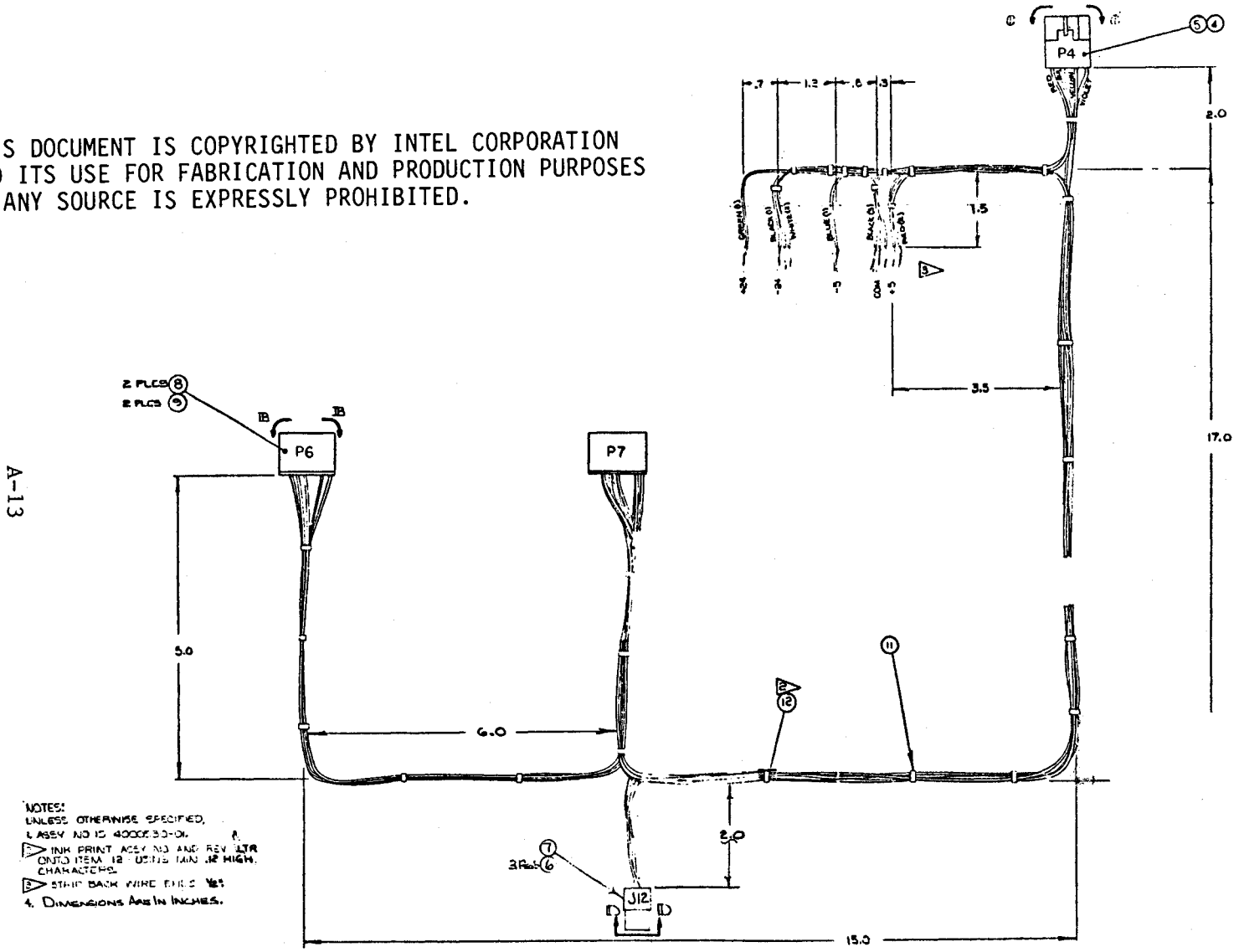
TITLE
HARNESS
FRONT PANEL

SIZE	DEPT	DRAWING NO.	REV
C	410	4000535	C



NOTES:
UNLESS OTHERWISE SPECIFIED.
1. ASSY NO IS 4001000-01.
2. INK STAMP APPROX WHERE SHOWN ASSY NO AND REV LTR USING MIN .12 HIGH CHARACTERS.
3. INSTALL CONNECTOR P11 ONTO FLAT CABLE APPROX 7½" FROM END CONN. P10 AS SHOWN.
4. CRIMP BENT END OF TIE WRAP WITH LONG NOSE PLIERS SO CABLE WILL NOT SLIP. CUT EXCESS TIE WRAP OFF.
5. DIMENSIONS ARE IN INCHES.

intel	3065 BOWERS AVE. SANTA CLARA CALIF. 95051		
TITLE: FLOPPY DISK FLAT CABLE - D.C. SIGNAL			
SIZE: C	DEPT: 410	DRAWING NO: 4001000	REV: A



NOTES:
UNLESS OTHERWISE SPECIFIED,
1. ASSY NO IS 4000530-01.
2. INK PRINT ASSY NO AND REV LTR ONTO ITEM 12 USING .100 IN .12 HIGH CHARACTERS.
3. STRIP BACK WIRE ENDS .125.
4. DIMENSIONS ARE IN INCHES.

J12
5 - WHITE
2 - VIOLET
1 - YELLOW
VIEW D - D

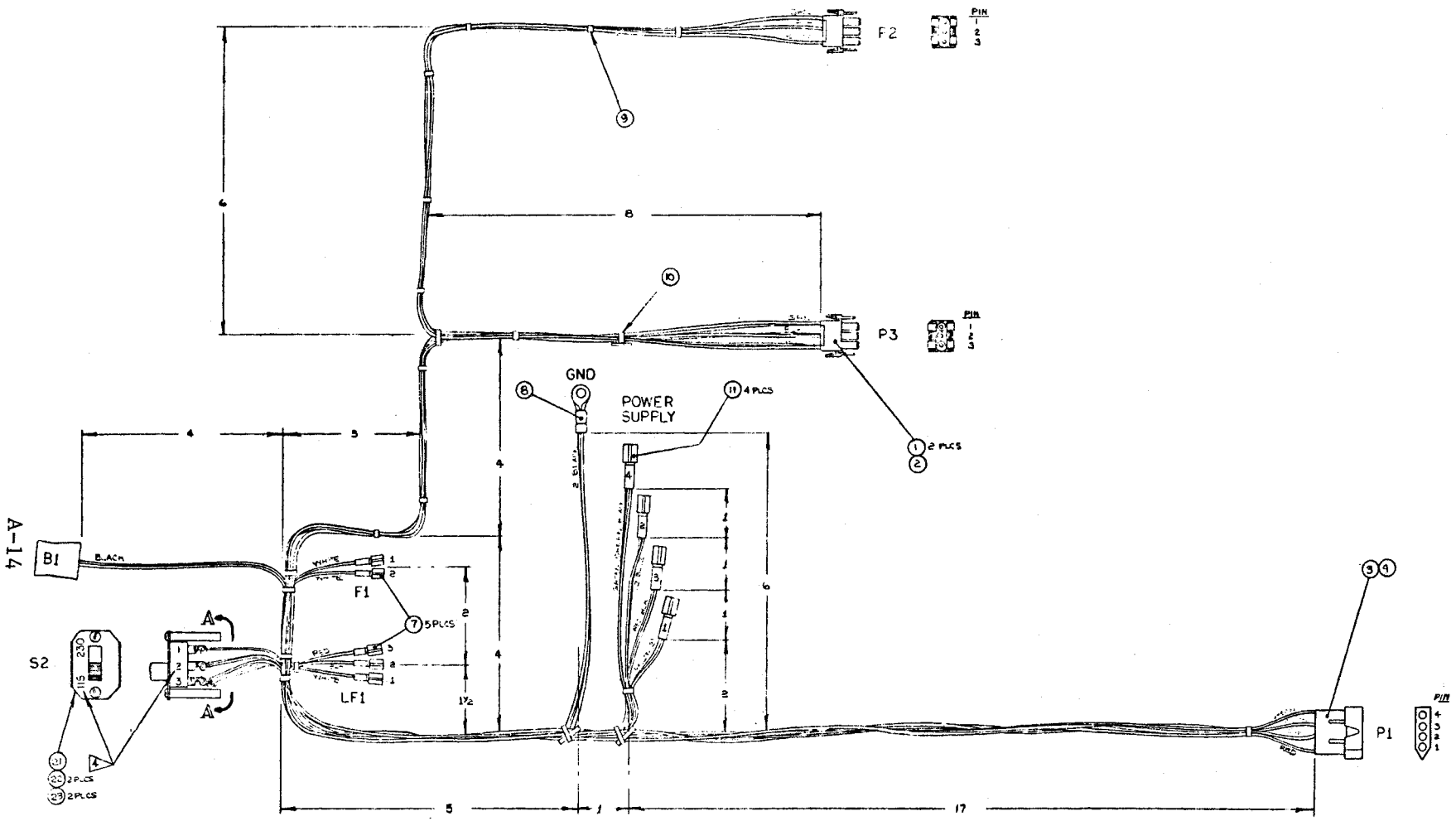
P4
4 - RED
3 - BLACK
2 - YELLOW
1 - VIOLET
VIEW C - C

P6, P7
8 (NOT USED)
7 - RED
6 - BLACK
5 - GREEN
4 (NOT USED)
3 - WHITE
2 - BLUE
VIEW B - B

intel 3065 BOWERS AVE. SANTA CLARA CALIF. 95051

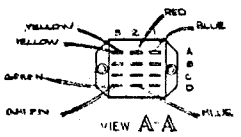
TITLE: HARNESS ASSEMBLY DCP II-EC

SIZE	DEPT	DRAWING NO.	REV
D	410	4000530	E

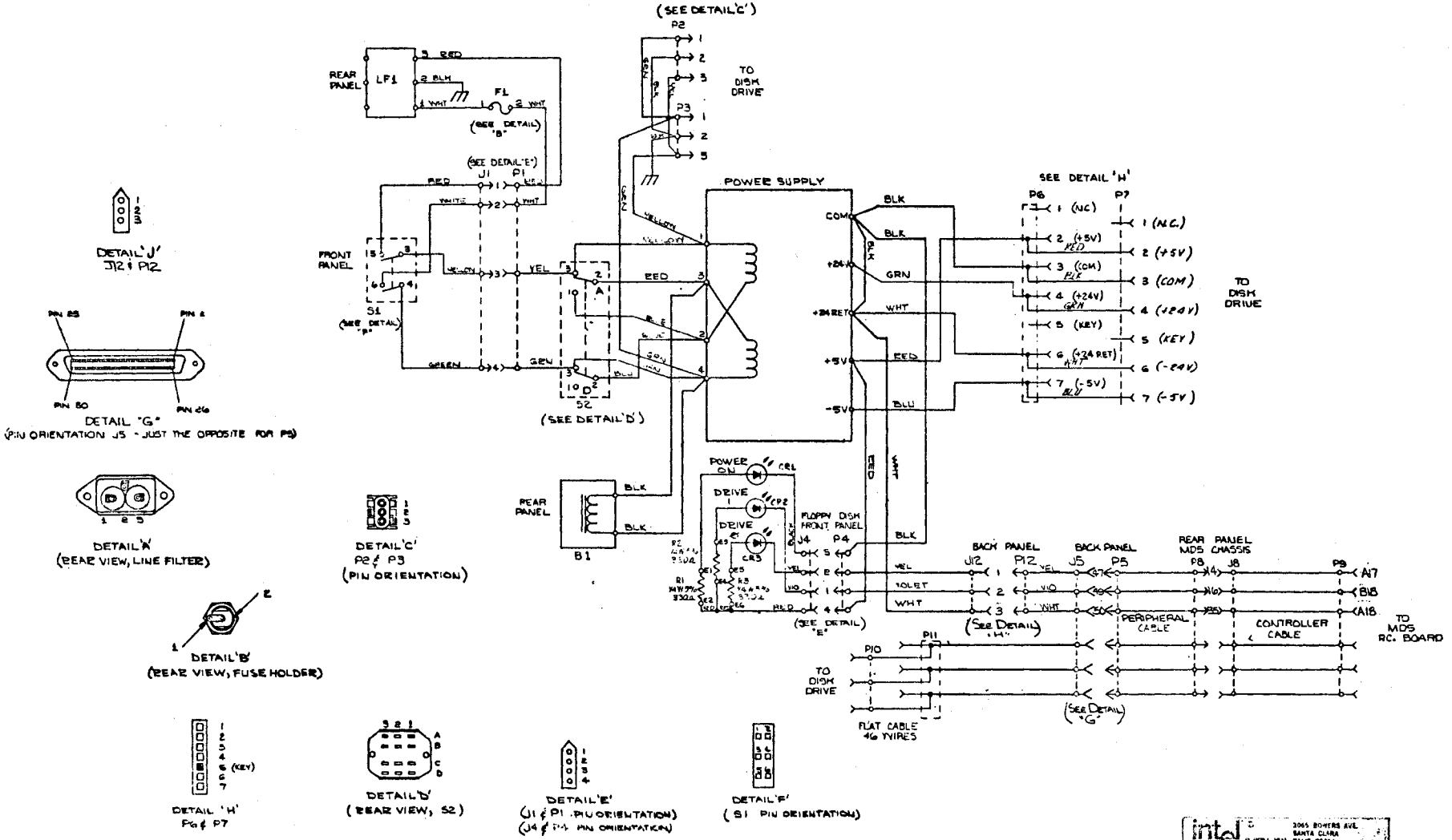


NOTES: UNLESS OTHERWISE SPECIFIED,

1. ASSY IS 4000529-01
2. PRINT ASSY NO & REV LETTER ONTO ITEM 10 USING MIN. .12 HIGH CHARACTERS.
3. DIMENSIONS ARE IN INCHES.
4. INK STAMP NOMENCLATURE IN ACCORDANCE EXACTLY AS SHOWN; CHARACTER HEIGHT TO BE MIN .12 HIGH, USING



DESCRIPTION			
PARTS LIST			
intel	D	3065 BOWERS AVE. SANTA CLARA CALIF. 95051	
HARNESS A.C. POWER			
SIZE	DEPT	DRAWING NO.	REV
D		4000529	D



NOTES:
1. UNLESS OTHERWISE SPECIFIED.
2. 'P' DESIGNATION INDICATES A MORE MOVABLE CONNECTOR.
3. 'J' DESIGNATION INDICATES A MORE FIXED CONNECTOR.
4. RESISTANCE VALUES ARE IN OHMS.

intel 3065 BOWERS AVE. SANTA CLARA

SCHEMATIC, AC/DC POWER

DISK DRIVE 4500613-01

PHYSICAL
WEIGHT : 15 LBS MAX

ELECTRICAL
AC: 3 WIRE CENTER CONDUCTOR TO CHASSIS GROUND
SINGLE PHASE 100 VAC AT 50/60HZ/120VAC AT 60 HZ
DC : +24 V AT 2.0 AMP
+5V AT 1.5 AMP
-5V AT 0.15 AMP

CONNECTORS
AC: AMP MATE'N LOCK 3 PIN AMP P/N 1-480701-0
USING AMP CONTACT P/N 350552-1 (MALE),
DC: AMP MOD. I RIGHT ANGLE HEADER
7 POSITION, .156 CTR, AMP P/N 87258-7
SIGNAL: SCOTCHFLEX 50 PIN RIGHT ANGLE HEADER
P/N 3433-1002

DRIVE MOTOR
360 RPM ± 3.5%

POSITIONER
STEPPER MOTOR / LEAD SCREW

FUNCTIONAL
BIT RATE : 249,984 BPS
RECORDING MODE : MODIFIED MFM
INTERCONNECTION : DAISY CHAIN CONFIGURATION
SYSTEM FEATURES:
UNIT READY INTERRUPT
WRITE FAULT CHECK
WRITE PROTECT
POWER REDUCTION
LOW CURRENT SWITCHING

TIMING REQUIREMENTS
POSITIONING : TRACK TO TRACK 10 MSEC MIN/STEP
PLUS 10 MSEC SETTLE
READ DATA STABILIZATION : 500 USEC AFTER
A WRITE COMMAND

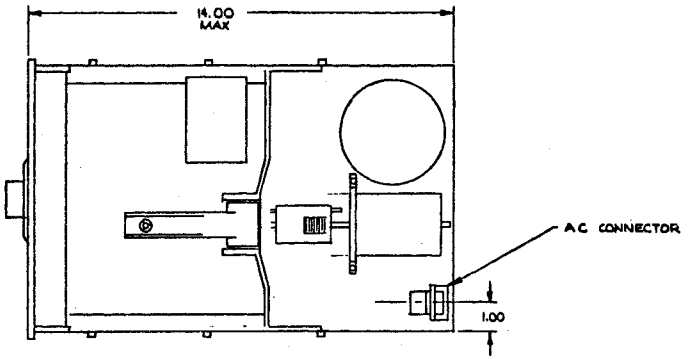
ELECTRICAL INTERFACE
TTL NEGATIVE TRUE LOGIC
DRIVERS TI SN 7438 OR EQUIV
RECEIVERS TI SN 75154 WITH A 220/330 TERMINATION

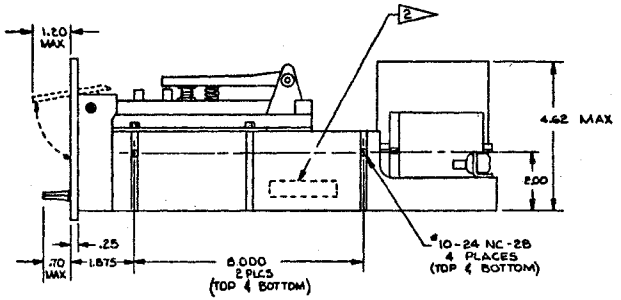
DISK DRIVE 4500613-02

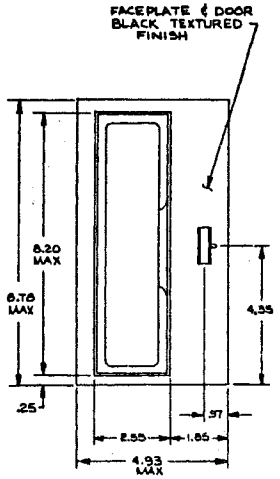
SAME AS 4500613-01 EXCEPT

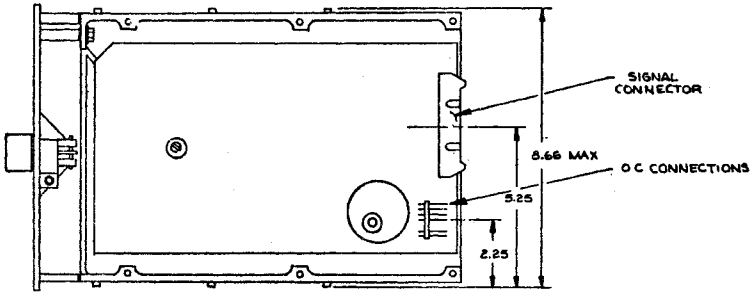
ELECTRICAL
AC: SINGLE PHASE 220 VAC AT 50 HZ

NOTE : UNLESS OTHERWISE SPECIFIED
1. PART NUMBER IS 4500613-XX
2. INK STAMP PART NO. AND REV LETTER IN POSITION SHOWN USING CONTRASTING INK, CHARACTERS TO BE .12 HIGH
3. FOR PROCUREMENT SEE L.V. 4500613.









DESCRIPTION			
PARTS LIST			
intel © INTEL 1978	3065 BOWERS AVE. SANTA CLARA CALIF. 95051		
TITLE: DISK DRIVE A			
SIZE: D	DEPT: 410	DRAWING NO.: 4500613	REV: C

ELECTRICAL SPECS :

A.C. INPUT
105-125 VAC OR 210-250 VAC
47-63 HZ SINGLE PHASE
PRIMARY TERMINATION IS TRANSFMR TERMINALS
VOLTAGE CHANGE OF PRIMARY IS BY TRANSFMR TAPS

OUTPUTS
+ 5VDC 8 A
- 5VDC 0.6A
+ 24VDC 4 A

REGULATION
LOAD 0.1% +5 MV NL-FL
LINE ± 0.1% ± 5 MV FOR ± 10% INPUT VARIATION

RIPPLE
2 MV RMS AND 20 MV P-P MAX.

TEMPERATURE COEFFICIENT
0.02%/°C MAX.

TEMPERATURE
OPERATING: 0° TO +65°C
STORAGE : -55° TO +85°C

TRANSIENT RESPONSE
OUTPUT VOLTAGE RETURNS TO WITHIN REGULATION LIMITS WITHIN 50 MICROSECONDS IN RESPONSE TO A 50% STEP CHANGE IN LOAD CURRENT FROM HALF TO FULL LOAD.

OVERLOAD PROTECTION
ELECTRONICALLY PROTECTED AGAINST OVERLOAD AND SHORT CIRCUITS OF ANY DURATION BY AUTOMATIC RECOVERY CURRENT LIMITING WITH FOLDBACK. CURRENT LIMIT IS ADJUSTABLE AND FACTORY SET AT 120 % OF FULL LOAD RATING.

OVERVOLTAGE PROTECTION
SCR "CROWBAR" TYPE PROTECTION. TRIP POINT IS ADJUSTABLE AND IS FACTORY SET ON 24 VOLT OUTPUT TO 27 VOLTS. ON THE +5VOLT OUTPUT A BRIDGE TYPE SINGLE "CROWBAR" WILL PROTECT BOTH OUTPUTS. TRIP POINT IS ADJUSTABLE AND IS FACTORY SET AT 12.5 V.

VOLTAGE ADJUSTMENT
+5V ± 0.25 VOLTS
-5V ± 0.25 VOLTS
+24V ± 1.0 VOLTS

TERMINATION
PRIMARY - TRANSFORMER TERMINALS
OUTPUT - SOLDER TERMINALS

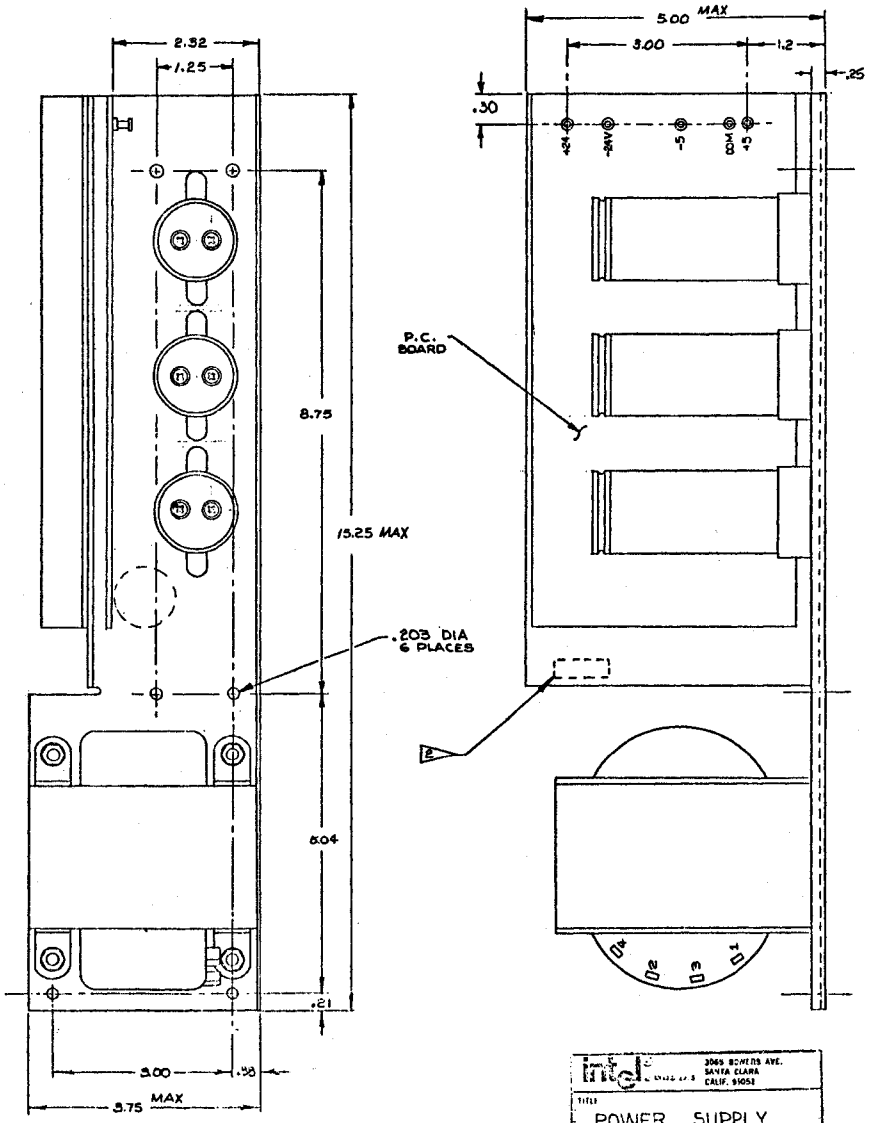
UNDERWRITERS LABORATORY
DESIGNED TO COMPLY WITH THE REQUIREMENTS OF UL478 FOR USE IN EDP EQUIPMENT.

COOLING
ENVIRONMENT WILL REQUIRE FORCED AIR OF APPROXIMATELY 20 CFM HORIZONTAL FLOW

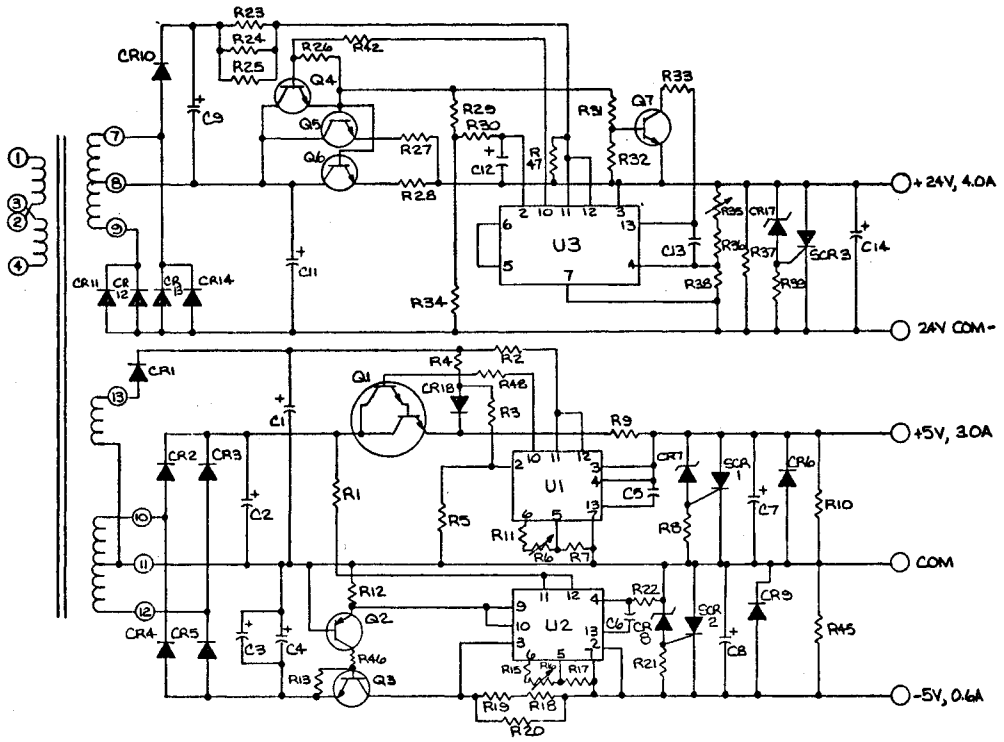
HUMIDITY
10% TO 90% NON-CONDENSING

MECHANICAL
OPEN FRAME CONSTRUCTION

NOTE: UNLESS OTHERWISE SPECIFIED
1. PART NO. IS 4500527-01
2. INK STAMP PART NO. AND REV LETTER IN POSITION SHOWN USING CONTRASTING INK, CHARACTERS TO BE .12 HIGH.
3. FOR PROCUREMENT SEE L.V. 4500527-01



intel 3065 BOWERS AVE. SANTA CLARA CALIF. 95052			
TITLE: POWER SUPPLY			
SIZE: D	DEPT: 410	DRAWING NO: 4500527	REV: F



C1,14	100/35	CAPACITOR ALUM ELECT
C2	9000/15	
C3,4	1000/16	
C7,8	220/16	
C9,12	47/50	
C11	9000/50	ALUM. ELECT
C5,6,13	.0039/100	CAPACITOR, MYLAR
CR1,4,5,6,9,10,16	AE1C	DIODE 1A 200V
CR2,3,11,12,13,14	AE3B	3A 100V
CR7,8	iN752A	ZENER
CR17	IN9718	DIODE ZENER,
Q1	2N4055	TRANSISTOR
Q2	2N2907A	
Q4,3	12500-5	
Q5,6	12509	TRANSISTOR
Q7	2N2219	TRANSISTOR
U1,2,3	μA723	I.C VOLTAGE REGULATOR
SCR1,2	S0303LS3	SCR 3A
SCR3	S0508LS3	SCR 8A
R1,8,21,46	47Ω	RESISTOR ½W 5% C.F.
R2,3,9,11,15,30,45	220Ω	
R4,12,13,22,26,32,37	2.2K	
R20,24,48	27Ω	
R30,31	10K	
R5	2.4K	
R29	750Ω	½W 5% C.F.
R7,17,36	2.3K	½W 5% M.F.
R18,19	.5Ω	½W 5% C.F.
R9	.12Ω	2W WW BWH
R27,28	.22Ω	2W WW BWH
R38,47	1.2K	RESISTOR ½W 5% M.F.
R6,16,35	1.5K	POTENTIOMETER
R23,24,25,32	7.5K	RESISTOR ½W 5% C.F.
R34	22K	RESISTOR ½W 5% C.F.
T1	12751	TRANSFORMER
P.C.B.	12780	PRINTED CIRCUIT BOARD
CHASSIS	12439	CHASSIS

DESCRIPTION

PARTS LIST

NOTE:
1 FOR VENDOR IDENTIFICATION SEE LV 4500527.

intel
3065 BOWERS AVE.
SANTA CLARA
CALIF. 95051

TITLE: SCHEMATIC, VIC AA POWER SUPPLY

SIZE	DEPT	DRAWING NO.	REV
D	410	2001045	A

APPENDIX B

FLEXIBLE DISKETTE CONTROLLER DRIVER EXAMPLE

```
PL/M-80 COMPILER

ISIS-II PL/M-80 V1.0 COMPILATION OF MODULE DISK
OBJECT MODULE PLACED IN :F1:DISK.OBJ
COMPILER INVOKED BY:  PLM80 :F1:DISK.MOD CODE


   1          DISK:
                                                ; STATEMENT # 1
              DO;
               /* ***************************************************************
                  ***************************************************************

                                       MODULE NAME DISK
                                       ====== ==== ====

                  ABSTRACT
                  ========

                      THIS MODULE IMPLEMENTS 4 PROCEDURES, EACH OF WHICH
                  ACCOMPLISHES THE TRANSFER OF 128 BYTES FROM/TO DISK TO/FROM
                  RAM MEMORY.

                  MODULE ORGANIZATION
                  ====== ============

                      SEVERAL 'LITERALLY' DEFINITIONS ARE GIVEN, FOLLOWED BY
                  THE CODE OF THE 4 PROCEDURES.

                  CALLING GRAPH
                  ======= =====

                  >>ABSREAD
                      'ABSIO'
                  >>ABSWRITE
                      'ABSIO'
                  >>ABSIO
                      'DISKIO'
                  >>DISKIO
                      ERR (ERROR)

                  GLOBAL VARIABLES ACCESSED
                  ====== ========= ========

                   -> FDCC$ERROR$TYPE (ERROR)

                  GLOBAL VARIABLES MODIFIED
                  ====== ========= ========

                   -> FDCC$ERROR$TYPE (ERROR)
               */

               /* ***************************************************************

                                       PROCEDURE DISKIO
                                       --------- ------

                  ABSTRACT
                  --------

                      THIS PROCEDURE PROVIDES ACCESS TO THE FDCC CONTROLLER(S),
```

```
                    CONTROLLER 1 = BASE ADDRESS 78H (DRIVES 0 AND 1),
                    CONTROLLER 2 = BASE ADDRESS 88H (DRIVES 2 AND 3).

                PARAMETERS
                ----------

                DRIVE1  AN INTEGER 0 THROUGH 3, SPECIFYING THE DISK TO BE
                        ACCESSED.
                IOPB    THE ADDRESS OF A PARAMETER BLOCK TO BE SENT TO THE
                        FDCC CONTROLLER.  THIS PARAMETER BLOCK MUST BE SET UP
                        AS IF IT WERE FOR DRIVE 0; IF 'DRIVE1' SELECTS
                        ANOTHER DRIVE, 'DISKIO' WILL SET ALL THE NECESSARY
                        ADDITIONAL BITS.

                VALUE RETURNED
                ----- --------

                NONE

                DESCRIPTION
                -----------

                    THE CALLER PROVIDES A PARAMETER BLOCK SPECIFYING SOME
                VALID DISK OPERATION ON DRIVE 0, AND AN INTEGER DRIVE
                SELECT VALUE.
                    THIS PROCEDURE WAITS FOR THE CONTROLLER TO GO UNBUSY, THEN
                PERFORMS THE DESIRED ACTION.  IF CASE OF A CONTROLLER ERROR,
                THE DISK DRIVE IS RECALIBRATED AND THE ACTION IS TRIED AGAIN.
                IF A SUCCESSFUL COMPLETION CANNOT BE OBTAINED AFTER "MAX$RETRIES"
                ATTEMPTS, A FATAL ERROR OCCURS; OTHERWISE A NORMAL RETURN IS
                MADE.
             */
            $NOLIST
                                                    ; STATEMENT # 2
                                                    ; STATEMENT # 3
                                                    ; STATEMENT # 4
                                                    ; STATEMENT # 5
                                                    ; STATEMENT # 6
                                                    ; STATEMENT # 7
                                                    ; STATEMENT # 8
                                                    ; STATEMENT # 9
                                                    ; STATEMENT # 10
                                                    ; STATEMENT # 11
                                                    ; STATEMENT # 12
                                                    ; STATEMENT # 13
                                                    ; STATEMENT # 14
                                                    ; STATEMENT # 15
                                                    ; STATEMENT # 16
                                                    ; STATEMENT # 17
                                                    ; STATEMENT # 18
                                                    ; STATEMENT # 19
                                                    ; STATEMENT # 20
                                                    ; STATEMENT # 21
                                                    ; STATEMENT # 22
                                                    ; STATEMENT # 23
                                                    ; STATEMENT # 24
                                                    ; STATEMENT # 25

26   1      DISKIO:
                                                    ; STATEMENT # 26
```

```
                        ; PROC  DISKIO
          0000  210200          LXI     H,IOPB+1H
          0003  72              MOV     M,D
          0004  2B              DCX     H
          0005  73              MOV     M,E
          0006  2B              DCX     H
          0007  71              MOV     M,C
              PROCEDURE (DRIVE,IOPB) PUBLIC;
  27    2       DECLARE DRIVE BYTE;
                                                ; STATEMENT # 27
  28    2       DECLARE TEMP ADDRESS, (TEMP2,TEMP1) BYTE AT ( TEMP);
                                                ; STATEMENT # 28
  29    2       DECLARE IOPB ADDRESS;
                                                ; STATEMENT # 29
  30    2       DECLARE DCB BASED IOPB STRUCTURE (
                                                ; STATEMENT # 30
                        IOCW BYTE,
                        IOINS BYTE,
                        NSEC BYTE,
                        TADR BYTE,
                        SADR BYTE,
                        BUF ADDRESS);
  31    2       DECLARE RECAL$PB STRUCTURE (
                                                ; STATEMENT # 31
                        IOCW BYTE,
                        IOINS BYTE,
                        NSEC BYTE,
                        TADR BYTE,
                        SADR BYTE);
  32    2       DECLARE I BYTE;
                                                ; STATEMENT # 32
  33    2       DECLARE IVAL BYTE;
                                                ; STATEMENT # 33

  34    2     DISK$STAT:
                                                ; STATEMENT # 34
                        ; PROC  DISKSTAT
              PROCEDURE BYTE;

  35    3         IF ROR(DRIVE,1) THEN
                                                ; STATEMENT # 35
          00D9  3A0000          LDA     DRIVE
          00DC  0F              RRC
          00DD  1F              RAR
          00DE  D2E400          JNC     @1
  36    3         DO;
                                                ; STATEMENT # 36
  37    4           RETURN INPUT(FDCC$STATUS$1);
                                                ; STATEMENT # 37
          00E1  DB88            IN      88H
          00E3  C9              RET
  38    4         END;
                                                ; STATEMENT # 38
                        @1:
  39    3         RETURN INPUT(FDCC$STATUS$0);
                                                ; STATEMENT # 39
          00E4  DB78            IN      78H
          00E6  C9              RET
  40    3       END DISK$STAT;
                                                ; STATEMENT # 40
```

```
41   2     R$TYPE:
                                                    ; STATEMENT # 41
                        ; PROC  RTYPE
           PROCEDURE BYTE;

42   3       IF ROR(DRIVE,1) THEN
                                                    ; STATEMENT # 42
           00E7  3A0000          LDA     DRIVE
           00EA  0F              RRC
           00EB  1F              RAR
           00EC  D2F200          JNC     @2
43   3       DO;
                                                    ; STATEMENT # 43
44   4         RETURN INPUT(RESULT$TYPE$1);
                                                    ; STATEMENT # 44
           00EF  DB89            IN      89H
           00F1  C9              RET
45   4       END;
                                                    ; STATEMENT # 45
                        @2:
46   3       RETURN INPUT(RESULT$TYPE$0);
                                                    ; STATEMENT # 46
           00F2  DB79            IN      79H
           00F4  C9              RET
47   3     END R$TYPE;
                                                    ; STATEMENT # 47

48   2     R$BYTE:
                                                    ; STATEMENT # 48
                        ; PROC  RBYTE
           PROCEDURE BYTE;

49   3       IF ROR(DRIVE,1) THEN
                                                    ; STATEMENT # 49
           00F5  3A0000          LDA     DRIVE
           00F8  0F              RRC
           00F9  1F              RAR
           00FA  D20001          JNC     @3
50   3       DO;
                                                    ; STATEMENT # 50
51   4         RETURN INPUT(RESULT$BYTE$1);
                                                    ; STATEMENT # 51
           00FD  DB88            IN      88H
           00FF  C9              RET
52   4       END;
                                                    ; STATEMENT # 52
                        @3:
53   3       RETURN INPUT(RESULT$BYTE$0);
                                                    ; STATEMENT # 53
           0100  DB78            IN      78H
           0102  C9              RET
54   3     END R$BYTE;
                                                    ; STATEMENT # 54

55   2     START$IO:
                                                    ; STATEMENT # 55
                        ; PROC  STARTIO
           0103  210D00          LXI     H,IOPB+1H
           0106  70              MOV     M,B
```

```
          0107  2B              DCX     H
          0108  71              MOV     M,C
              PROCEDURE (IOPB);
  56   3        DECLARE IOPB ADDRESS;
                                                  ; STATEMENT # 56

  57   3        IF ROR(DRIVE,1) THEN
                                                  ; STATEMENT # 57
          0109  3A0000          LDA     DRIVE
          010C  0F              RRC
          010D  1F              RAR
          010E  D21D01          JNC     @4
  58   3        DO;
                                                  ; STATEMENT # 58
  59   4          OUTPUT(LOW$ADDRESS$1) = LOW(IOPB);
                                                  ; STATEMENT # 59
          0111  2A0C00          LHLD    IOPB
          0114  7D              MOV     A,L
          0115  D389            OUT     89H
  60   4          OUTPUT(HIGH$ADDRESS$1) = HIGH(IOPB);
                                                  ; STATEMENT # 60
          0117  7C              MOV     A,H
          0118  D38A            OUT     8AH
  61   4        END;
                                                  ; STATEMENT # 61
          011A  C32601          JMP     @5
                      @4:
                ELSE
  62   3        DO;
                                                  ; STATEMENT # 62
  63   4          OUTPUT(LOW$ADDRESS$0) = LOW(IOPB);
                                                  ; STATEMENT # 63
          011D  2A0C00          LHLD    IOPB
          0120  7D              MOV     A,L
          0121  D379            OUT     79H
  64   4          OUTPUT(HIGH$ADDRESS$0) = HIGH(IOPB);
                                                  ; STATEMENT # 64
          0123  7C              MOV     A,H
          0124  D37A            OUT     7AH
  65   4        END;
                                                  ; STATEMENT # 65
                      @5:
  66   3      END START$IO;
                                                  ; STATEMENT # 66
          0126  C9              RET

                /* PREVENT INTERRUPT 1 FROM CAUSING REENTRANCY. */

  67   2        IVAL = INPUT(0FCH);
                                                  ; STATEMENT # 67
          0008  DBFC            IN      0FCH
          000A  320B00          STA     IVAL
  68   2        OUTPUT(0FCH) = IVAL OR 2; /* MASK LEVEL 1, LEAVE REST UNCHANGED */
                                                  ; STATEMENT # 68

          000D  3A0B00          LDA     IVAL
          0010  F602            ORI     2H
          0012  D3FC            OUT     0FCH
  69   2        RECAL$PB.IOCW = 80H;
                                                  ; STATEMENT # 69
```

```
                0014  210500          LXI     H,RECALPB
                0017  3680            MVI     M,80H
  70    2           RECAL$PB.IOINS = RECALIBRATE;
                                                ; STATEMENT # 70
                0019  23              INX     H
                001A  3603            MVI     M,3H
  71    2           RECAL$PB.SADR = 0;
                                                ; STATEMENT # 71
                001C  210900          LXI     H,RECALPB+4H
                001F  3600            MVI     M,0H
  72    2           IF DRIVE THEN
                                                ; STATEMENT # 72
                0021  3A0000          LDA     DRIVE
                0024  1F              RAR
                0025  D24500          JNC     @6
  73    2           DO;
                                                ; STATEMENT # 73
  74    3             DCB.IOINS = DCB.IOINS OR 30H;
                                                ; STATEMENT # 74
                0028  2A0100          LHLD    IOPB
                002B  23              INX     H
                002C  3E30            MVI     A,30H
                002E  B6              ORA     M
                002F  77              MOV     M,A
  75    3             DCB.SADR = DCB.SADR OR 20H;
                                                ; STATEMENT # 75
                0030  010400          LXI     B,4H
                0033  2A0100          LHLD    IOPB
                0036  09              DAD     B
                0037  3E20            MVI     A,20H
                0039  B6              ORA     M
                003A  77              MOV     M,A
  76    3             RECAL$PB.IOINS = RECALIBRATE OR 30H;
                                                ; STATEMENT # 76
                003B  210600          LXI     H,RECALPB+1H
                003E  3633            MVI     M,33H
  77    3             RECAL$PB.SADR = 20H;
                                                ; STATEMENT # 77
                0040  210900          LXI     H,RECALPB+4H
                0043  3620            MVI     M,20H
  78    3           END;
                                                ; STATEMENT # 78
                                @6:

  79    2           DO I = 0 TO MAX$RETRIES;
                                                ; STATEMENT # 79
                0045  210A00          LXI     H,I
                0048  3600            MVI     M,0H
                                @9:
                004A  3E0A            MVI     A,0AH
                004C  210A00          LXI     H,I
                004F  BE              CMP     M
                0050  DAC800          JC      @10
  80    3             DO WHILE (DISK$STAT AND DISK$DONE) <> 0;
                                                ; STATEMENT # 80
                                @11:
                0053  CDD900          CALL    DISKSTAT
                0056  E604            ANI     4H
                0058  FE00            CPI     0H
                005A  CA6C00          JZ      @12
```

```
81   4              TEMP1 = R$TYPE;
                                                    ; STATEMENT # 81
          005D  CDE700          CALL    RTYPE
          0060  320400          STA     TEMP1
82   4              TEMP1 = R$BYTE;
                                                    ; STATEMENT # 82
          0063  CDF500          CALL    RBYTE
          0066  320400          STA     TEMP1
83   4            END;
                                                    ; STATEMENT # 83
          0069  C35300          JMP     @11
                        @12:
                  /* IF DISK DRIVE NOT READY, GIVE FATAL ERROR */

84   3            IF (DISK$STAT AND ((DRIVE AND 1)+1)) = 0 THEN
                                                    ; STATEMENT # 84
          006C  CDD900          CALL    DISKSTAT
          006F  F5              PUSH    PSW     ; 1
          0070  3A0000          LDA     DRIVE
          0073  E601            ANI     1H
          0075  3C              INR     A
          0076  C1              POP     B       ; 1
          0077  48              MOV     C,B
          0078  A1              ANA     C
          0079  FE00            CPI     0H
          007B  C28500          JNZ     @7
85   3              CALL ERR(ABORT,DRIVE$NOT$READY);
                                                    ; STATEMENT # 85
          007E  1E1E            MVI     E,1EH
          0080  0E02            MVI     C,2H
          0082  CD0000          CALL    ERR
                        @7:
86   3            CALL START$IO(IOPB);
                                                    ; STATEMENT # 86
          0085  2A0100          LHLD    IOPB
          0088  44              MOV     B,H
          0089  4D              MOV     C,L
          008A  CD0301          CALL    STARTIO
87   3            DO WHILE (DISK$STAT AND DISK$DONE) = 0;
                                                    ; STATEMENT # 87
                        @13:
          008D  CDD900          CALL    DISKSTAT
          0090  E604            ANI     4H
          0092  FE00            CPI     0H
          0094  C29A00          JNZ     @14
88   4              ;
                                                    ; STATEMENT # 88
89   4            END;
                                                    ; STATEMENT # 89
          0097  C38D00          JMP     @13
                        @14:
90   3            TEMP1 = R$TYPE;
                                                    ; STATEMENT # 90
          009A  CDE700          CALL    RTYPE
          009D  320400          STA     TEMP1
91   3            IF (TEMP2 :=R$BYTE) = 0 THEN
                                                    ; STATEMENT # 91
          00A0  CDF500          CALL    RBYTE
          00A3  320300          STA     TEMP2
          00A6  FE00            CPI     0H
```

```
                00A8  C2B100          JNZ     @8
 92    3                DO;
                                                ; STATEMENT # 92
 93    4                  OUTPUT(OFCH) = IVAL; /* RESTORE INTERRUPT 1 */
                                                ; STATEMENT # 93
                00AB  3A0800          LDA     IVAL
                00AE  D3FC            OUT     OFCH
 94    4                  RETURN;
                                                ; STATEMENT # 94
                00B0  C9              RET
 95    4                END;
                                                ; STATEMENT # 95
                            @8:
 96    3                CALL START$IO(.RECAL$PB);
                                                ; STATEMENT # 96
                00B1  010500          LXI     B,RECALPB
                00B4  CD0301          CALL    STARTIO
 97    3                DO WHILE (DISK$STAT AND DISK$DONE) = 0;
                                                ; STATEMENT # 97
                            @15:
                00B7  CDD900          CALL    DISKSTAT
                00BA  E604            ANI     4H
                00BC  FE00            CPI     0H
                00BE  C2C400          JNZ     @16
 98    4                  ;
                                                ; STATEMENT # 98
 99    4                END;
                                                ; STATEMENT # 99
                00C1  C3B700          JMP     @15
                            @16:
100    3                END;
                                                ; STATEMENT # 100
                00C4  210A00          LXI     H,I
                00C7  34              INR     M
                00C8  C24A00          JNZ     @9
                            @10:

101    2                FDCC$ERROR$TYPE = TEMP;
                                                ; STATEMENT # 101
                00CB  2A0300          LHLD    TEMP
                00CE  220000          SHLD    FDCCERRORTYPE
102    2                CALL ERR(ABORT,DISKIO$ERROR);
                                                ; STATEMENT # 102
                00D1  1E18            MVI     E,18H
                00D3  0E02            MVI     C,2H
                00D5  CD0000          CALL    ERR
103    2            END DISKIO;
                                                ; STATEMENT # 103
                00D8  C9              RET

                   /* ***************************************************************

                                        PROCEDURE ABSIO
                                        --------- -----

                      ABSTRACT
                      --------

                          ABSIO ACCOMPLISHES THE TRANSFER OF 128 BYTES OF
                      DATA TO/FROM THE DISKETTE.
```

```
                PARAMETERS
                ----------
                COMMAND      MUST BE THE NUMERIC VALUE OF THE FDCC COMMAND
                             DESIRED.  (LITERALS 'READ$COMMAND' AND 'WRITE$COMMAND'
                             EXIST FOR THE COMMON OPERATONS.)
                DISK         INTEGER 0 OR 1, SELECTS WHICH DRIVE.
                BLOCK        DISKETTE BLOCK NUMBER, A TRACK NUMBER (0-76)
                             IN THE HIGH ORDER 8 BITS AND A SECTOR NUMBER (1-26)
                             IN THE LOW ORDER 8 BITS.
                BUFFER$PTR   THE ADDRESS OF A 128 BYTE BUFFER IN RAM.

                VALUE RETURNED
                ----- --------

                NONE

                DESCRIPTION
                -----------

                AN I/O PARAMETER BLOCK ("DCB") IS SETUP ACCORDING TO THE
                PARAMETERS PROVIDED, AND "DISKIO" IS CALLED.

              */

104   1       ABSIO:
                                                        ; STATEMENT # 104
                            ; PROC  ABSIO
          0127  211300              LXI     H,BUFFERPTR+1H
          012A  72                  MOV     M,D
          012B  28                  DCX     H
          012C  73                  MOV     M,E
          012D  28                  DCX     H
          012E  70                  MOV     M,B
          012F  28                  DCX     H
          0130  71                  MOV     M,C
          0131  28                  DCX     H
          0132  D1                  POP     D
          0133  C1                  POP     B
          0134  71                  MOV     M,C
          0135  28                  DCX     H
          0136  C1                  POP     B
          0137  71                  MOV     M,C
          0138  D5                  PUSH    D
              PROCEDURE (COMMAND,DISK,BLOCK,BUFFER$PTR) PUBLIC;
105   2           DECLARE (COMMAND,DISK) BYTE;
                                                        ; STATEMENT # 105
106   2           DECLARE (BLOCK,BUFFER$PTR) ADDRESS;
                                                        ; STATEMENT # 106
                  /* VALUE OF 'DISK' MUST BE 0  - 3 */
107   2           DECLARE DCB STRUCTURE (
                                                        ; STATEMENT # 107
                          IOCW BYTE,
                          IOINS BYTE,
                          NSEC BYTE,
                          TADR BYTE,
                          SADR BYTE,
                          BUF ADDRESS);

108   2           DCB.IOCW = 80H;
```

```
                                                    ; STATEMENT # 108
          0139  211400          LXI     H,DCB
          013C  3680            MVI     M,80H
109   2         DCB.IOINS = COMMAND;
                                                    ; STATEMENT # 109
          013E  3A0E00          LDA     COMMAND
          0141  23              INX     H
          0142  77              MOV     M,A
110   2         DCB.NSEC = 1;
                                                    ; STATEMENT # 110
          0143  23              INX     H
          0144  3601            MVI     M,1H
111   2         DCB.TADR = HIGH(BLOCK);
                                                    ; STATEMENT # 111
          0146  2A1000          LHLD    BLOCK
          0149  7C              MOV     A,H
          014A  321700          STA     DCB+3H
112   2         DCB.SADR = LOW(BLOCK);
                                                    ; STATEMENT # 112
          014D  7D              MOV     A,L
          014E  321800          STA     DCB+4H
113   2         DCB.BUF = BUFFER$PTR;
                                                    ; STATEMENT # 113
          0151  2A1200          LHLD    BUFFERPTR
          0154  221900          SHLD    DCB+5H
114   2         CALL DISKIO(DISK,.DCB);
                                                    ; STATEMENT # 114
          0157  2A0F00          LHLD    DISK
          015A  4D              MOV     C,L
          015B  111400          LXI     D,DCB
          015E  CD0000          CALL    DISKIO
115   2       END ABSIO;
                                                    ; STATEMENT # 115
          0161  C9              RET

             /* ***************************************************************

                              PROCEDURES ABSREAD, ABSWRITE
                              ---------- -------- --------

                 ABSTRACT
                 --------

                 THESE PROCEDURES PROVIDE SHORTER CALLING SEQUENCES FOR
                 THE 2 MOST COMMON USES OF THE "ABSIO" PROCEDURE.

                 PARAMETERS
                 ----------

                 BLOCK       (SAME AS FOR "DISKIO", ABOVE.)
                 BUFFER$PTR  (SAME AS FOR "DISKIO", ABOVE.)

                 VALUE RETURNED
                 ----- --------

                 NONE

              */

116   1      ABSREAD:
```

```
                                                    ; STATEMENT # 116
                        ; PROC  ABSREAD
          0162  211E00          LXI     H,BUFFERPTR+1H
          0165  72              MOV     M,D
          0166  2B              DCX     H
          0167  73              MOV     M,E
          0168  2B              DCX     H
          0169  70              MOV     M,B
          016A  2B              DCX     H
          016B  71              MOV     M,C
             PROCEDURE(BLOCK,BUFFER$PTR) PUBLIC;
117   2        DECLARE(BLOCK,BUFFER$PTR) ADDRESS;
                                                    ; STATEMENT # 117

118   2        CALL ABSIO(READ$COMMAND,A$DEVICE,BLOCK,BUFFER$PTR);
                                                    ; STATEMENT # 118
          016C  0E04            MVI     C,4H
          016E  C5              PUSH    B       ; 1
          016F  2A0000          LHLD    ADEVICEBASE
          0172  4E              MOV     C,M
          0173  C5              PUSH    B       ; 2
          0174  2A1B00          LHLD    BLOCK
          0177  44              MOV     B,H
          0178  4D              MOV     C,L
          0179  2A1D00          LHLD    BUFFERPTR
          017C  E8              XCHG
          017D  CD2701          CALL    ABSIO
119   2        END ABSREAD;
                                                    ; STATEMENT # 119
          0180  C9              RET

120   1     ABSWRITE:
                                                    ; STATEMENT # 120
                        ; PROC  ABSWRITE
          0181  212200          LXI     H,BUFFERPTR+1H
          0184  72              MOV     M,D
          0185  2B              DCX     H
          0186  73              MOV     M,E
          0187  2B              DCX     H
          0188  70              MOV     M,B
          0189  2B              DCX     H
          018A  71              MOV     M,C
             PROCEDURE(BLOCK,BUFFER$PTR) PUBLIC;
121   2        DECLARE(BLOCK,BUFFER$PTR) ADDRESS;
                                                    ; STATEMENT # 121

122   2        CALL ABSIO(WRITE$COMMAND,A$DEVICE,BLOCK,BUFFER$PTR);
                                                    ; STATEMENT # 122
          018B  0E06            MVI     C,6H
          018D  C5              PUSH    B       ; 1
          018E  2A0000          LHLD    ADEVICEBASE
          0191  4E              MOV     C,M
          0192  C5              PUSH    B       ; 2
          0193  2A1F00          LHLD    BLOCK
          0196  44              MOV     B,H
          0197  4D              MOV     C,L
          0198  2A2100          LHLD    BUFFERPTR
          019B  EB              XCHG
          019C  CD2701          CALL    ABSIO
123   2        END ABSWRITE;
```

```
PL/M-80 COMPILER


                                                        ; STATEMENT # 123
          019F  C9                  RET
 124   1      END;
                                                        ; STATEMENT # 124

              EOF


MODULE INFORMATION:

     CODE AREA SIZE      = 01A0H     416D
     VARIABLE AREA SIZE  = 0023H      35D
     MAXIMUM STACK SIZE  = 0006H       6D
     490 LINES READ
     0 PROGRAM ERROR(S)

END OF PL/M-80 COMPILATION
```